The Thesis Committee for Andrew W. Karl
Certifies that this is the approved version of the following thesis:

Developing a GIS-Based Traffic Control Planning Tool

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor: ______________________________
Randy B. Machemehl

____
Zhanmin Zhang

Developing a GIS-Based Traffic Control Planning Tool

by

Andrew W. Karl, B.S.C.E.

Thesis

Presented to the Faculty of the Graduate School
of The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Masters of Science in Engineering

The University of Texas at Austin
December 2009

Dedication

My family

Acknowledgements

I would like to thank Dr. Machemehl for his help and guidance during my time spent working on this project. I thank Dr. Zhang for being my second reader. Thank you to Dongliang Xu for his hard work on the complex programming that went into developing the tools for this project. Thank you to Katherine Kortum for her help in gathering, organizing, and analyzing the large amounts of data we had to deal with. Special thanks to Rui Gao for taking the time to train me on how to use GIS and continue work on this project.

Abstract

Developing a GIS-Based Traffic Control Planning Tool

by

Andrew W. Karl, MSE
The University of Texas at Austin, 2009
Supervisor: Randy B. Machemehl

The purpose of this study is to assist TxDOT engineers in the field of traffic control planning. This is to be done via the creation of a Geographic Information System (GIS) based tool. By bringing together information about TxDOT's on-system roadways' geographical locations, traffic demands, and capacities, one aggregate database has been established. Using the tools of GIS, Microsoft Excel, Microsoft Access, and VBA programming, a static clickable interface has been constructed. It enables users to access properties for any selected roadway link they desire. Expansion of the product to ArcIMS is ongoing to allow easy access for end users via the internet.

Table of Contents

List of Tables

List of Figures

List of Instructions

List of Programs

Chapter 1 – Introduction

1.1 MOTIVATION FOR THE WORK

Transportation is key to economic development and maintaining people's quality of life. A strong transportation network allows users to access everything they need to carry on with daily activities whether it be getting to work, visiting friends and family, seeing a doctor, or purchasing basic goods. However, the transportation infrastructure in the United States has been under great pressure in recent years. The combination of record numbers of users and aging transport networks has led to costly dilemmas for transportation authorities nationwide. "As with other major infrastructure systems that support society…the importance of the nation's transportation system becomes apparent only when problems arise" (TRB, 2006). This holds especially true in urban areas such as Dallas, TX. In 2007, delay per traveler in the Dallas - Fort Worth region was about 53 hours annually, ranking the region sixth nationally in that category (Lomax & Schrank, 2009). The only way to alleviate such problems is through continual and proper maintenance of the existing infrastructure and construction of better facilities.

Decisions on rehabilitation and upgrades, which may involve lane closures, must take into account many factors. These include the condition of the infrastructure (i.e., pavement, bridges, tunnels, etc.), traffic demand on the roadway, and impact on the surrounding environment. Something that is often not immediately thought of is user cost. These costs are exemplified by the extra time it takes people to travel along detours around their normal routes and, as an example, how much extra gas will be used and the resulting increase in vehicle emissions. It is difficult to account for these somewhat “invisible” costs during the traffic control planning process. Fortunately, traffic engineers have a resource to guide them when devising temporary traffic control.

The Manual on Uniform Traffic Control Devices (MUTCD) is a product of the Federal Highway Administration (FHWA) which “defines the standards used by road managers nationwide to install and maintain traffic control devices on all streets and highways” (MUTCD, 2003). Part 6 of the MUTCD covers Temporary Traffic Control, or TTC. It is in this section that they present an overview of plans for different scenarios including signage, markings, and flaggers, among other things. “The needs and control of all road users…through a TTC zone shall be an essential part of highway construction, utility work, maintenance operations, and the management of traffic incidents” (MUTCD, 2003). Work zones often experience constant changes in conditions both for the drivers and workers. It is imperative that all guidelines needed for motorists, cyclists, and pedestrians be made very visible and easy to understand. “Consideration for road user safety, worker and responder safety, and the efficiency of road user flow is an integral element of every TTC zone, from planning through completion” (MUTCD, 2003). It is very important that the TTC process and evaluation be ongoing throughout the project and not be limited to the planning stage alone. The overall development of a TTC is very much tailored to the site characteristics. That is why it is important to supply these developing traffic control plans with as much information as possible including roadway characteristics, roadway capacity, and traffic demand. For the Dallas district of TxDOT, these three sets of data are in separate databases.

1.2 WORK METHODOLOGY AND OBJECTIVES

The main goal of this project is to create a static model with a clickable interface which will display the demand on and capacity of TxDOT on-system roadways. In order to create the GIS-Based Traffic Control Planning Tool, it is necessary to have three particular sets of information. These sets are: the actual geographical layout of the road network, the traffic demand on these roads, and the capacity of the roads.

The geographical layout of Texas Department of Transportation (TxDOT) on-system roadways is stored in a GIS format for each county. This GIS database contains information on the district responsible for each roadway. It has no details concerning roadway capacity or demand. From this point forward, this database will be referred to as Census Roads. The network is stored in a shapefile containing vector lines representing roads.

The capacity of TxDOT's on-system roads is calculated from characteristics such as lane width and shoulder width, among others. For this project, such information is obtained from a TxDOT database known as the Pavement Management Information System (PMIS). The database contains hundreds of categories of information needed to identify the state of the roads and determine when maintenance is necessary. The PMIS is only a database, but it is linked to geographical locations through the Texas Reference Marker (TRM) system. The TRM system will be discussed more in Section 3.4.

Current traffic volumes on TxDOT roadways would ideally come from current traffic counts. Although large numbers of traffic counts are available, they do not provide volumes for every segment of every facility and they lack a common time reference. Due to this, demand numbers are taken from the North Central Texas Council of Governments' (NCTCOG) Dallas – Fort Worth Regional Travel Model (DFWRTM). It is a four-step traffic demand forecasting model that uses a static traffic assignment procedure. This assignment procedure produces traffic demand that is numerically different from counted volumes and requires conversion to estimate the needed volume count data. This modeling procedure will be discussed further in Chapter 2. The traffic volume is defined as the number of vehicles using a roadway during a peak hour time period, or the sum of four consecutive 15-minute intervals which produce the highest volume.

Unfortunately, the idea of combining all three sets of data is not as simple as was initially hoped. These databases are from different sources and do not contain a uniform identifier. There was no one attribute to look at and use to link all three sets of data together. Another problem is that each database is in a different coordinate system so that the layers do not line up with one another properly. The last difficulty is that each database breaks down the roadways into variously sized segments, with no discernible pattern. A stretch of highway may be one continuous segment in Census Roads, but could be broken into as many as ten segments in the Demand database. Additionally, the PMIS information is already in half-mile increments of its own. Creating uniformity was a challenge in itself. Chapter 5 will discuss how these problems were solved. Figure 1.1 demonstrates how the layers do not match.

Figure 1.1: Comparison of Census Roads Layer and Demand layer

1.3 STUDY AREA

The study area for this project is the Dallas District of TxDOT. This includes the following seven counties: Collin, Dallas, Denton, Ellis, Kaufman, Navarro, and Rockwall. The city of Fort Worth is not included in the Dallas district and is not accounted for in this study. The DFWRTM does not cover Navarro County. Other methods will need to be sought to acquire demand values on those roadways.

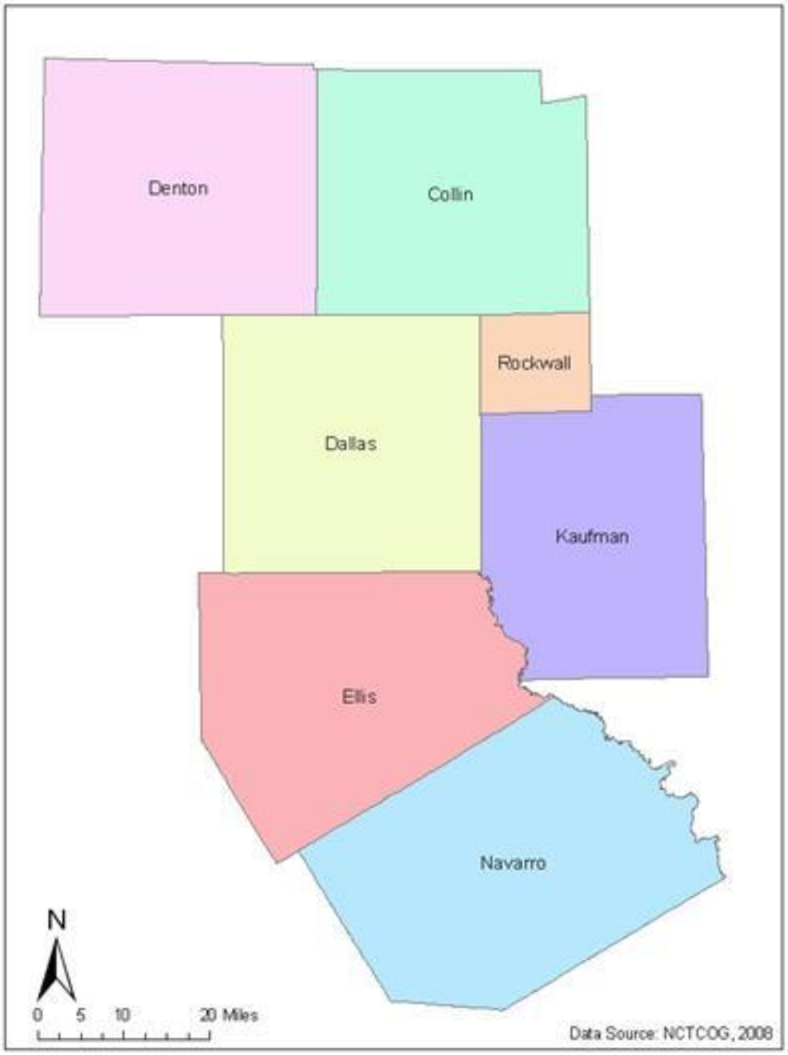


Figure 1.2: TxDOT's Dallas District Counties

1.4 THESIS STRUCTURE

This project is scheduled for three separate phases of development. Phase 1 entails the creation of the static model by combining the census roads, capacity calculations, and demand numbers. This includes the development of a tool which allows the user to click on a section of roadway and then click on the tool to generate a pop-up display window providing information on that segment. Such information would include roadway name, capacity, and demand. This tool will also allow the user to edit certain roadway facets (i.e. lane width) and then recalculate the capacity.

Phase 2 calls for the model to be put into a traffic microsimulator such as CORSIM or VISSIM which will generate further information for enhancing traffic flow. It is an integral piece to the project. If a lane on a highway needs to be closed due to an accident or construction, the simulator allows an engineer to see how traffic will back up behind the lane closure. It will also provide an idea of how vehicles may exit a highway and distribute themselves throughout the network, possibly leading to traffic problems in other areas due to an overflow of vehicles.

Phase 3 involves assimilating the traffic assignment step of the regional demand estimation model and desirably substituting dynamic traffic assignment for the current static traffic assignment process. This will turn what is now a static model into a dynamic model. It would open up the tool to live updates of the most recent traffic forecasts and ensure the user is not working with outdated numbers.

This thesis will focus on Phase 1 of the project. Chapter 2 involves a review of Geographic Information Systems and forecast modeling. Chapters 3 and 4 discuss capacity and demand. Chapter 5 covers the specific steps followed to combine the three separate databases. Chapters 6 and 7 look at the most up-to-date model and summarize the project thus far.

Chapter 2 – Literature Review

2.1 GEOGRAPHIC INFORMATION SYSTEMS

"The advent of geographic information systems was the result not of academic inquiry but rather of the growing societal need for geographical information, of a change in the technology that made such systems possible, and of private sector vision and government foresight that initiated and sustained their development" (Tomlinson, 1998). This quote concerns the dawn of the Canada Geographic Information System. During the 1960s, Canada had become concerned with the state of the country's natural resources. Faced with a need to manage land use, the government, with the aid of the commercial sector, decided to inventory its resources. Such copious amounts of geographical data needed to be stored and analyzed quickly. The automation provided by computers was a perfect fit. The main piece of data being amassed by the GIS is location.

"Almost everything that happens, happens somewhere. Knowing where something happens can be critically important" (Longley, 2008). A GIS is a very powerful tool which empowers the user to amalgamate different types of information from separate sources and view them in a way that allows relationships and patterns to become clearly visible. It uses the spatial component of a set of data to relate things to one another in an interactive setting. Geographic issues face everyone in the world. How to travel to work or where to build a hospital are just a couple of examples of problems that a GIS can help solve. Another problem that can be solved is how to control traffic.

A GIS tool is a very appropriate solution to the task of creating traffic control plans (TCPs). It is first and foremost a geodatabase, storing large amounts of spatial information. It is capable of holding information on correct road layout, traffic demand, and capacity. A GIS can then map this information. Engineers need to know the exact layout of the road network they are dealing with. Which roads are access or non-access controlled? Are there signalized intersections affecting roadway capacity? What would be the best way to detour traffic around construction? These are all questions that, with the correct datasets, can be answered through the creation of a GIS tool.

The specific GIS package used in this project is called ArcView and was developed by ESRI.

2.2 DEMAND MODELING

The need for traffic demand modeling around the world was the result of population growth and the many problems it has caused. There are more people than ever needing a form of transport between their origins and destinations. Traffic congestion, vehicular accidents, and environmental pollution are just some of the negative effects of an overburdened transportation infrastructure. Managing these problems and searching for solutions are truly dubious tasks without the aid of some form of data processor. "However, electronics and computing have advanced so much as to make possible new conceptions of transport infrastructure…and movement systems…" (Ortuzar et al., 1990). There are many factors that affect traffic in an area including population growth, people's income or social class, schools in the community, and local retail facilities. In order to account for these and many other determinants affecting people's trip-making, a mathematical-based model was needed.

Today, many organizations across the country that are in charge of transportation forecasts for their region employ the Urban Transportation Model System (UTMS). It is a four-step model comprised of trip generation, trip distribution, mode split, and traffic assignment.

Trip generation is the initial step. The number of trips originating from a zone are determined by that zone's socioeconomic background. This includes factors like ethnic makeup and income among other things. However, this step does not account for changes to the transportation system or changes in area policy, both of which can have a major effect on trip generation. Another problem is the concept of synthetic population generation. Because the exact characteristics of everyone living in a study area are unknown, small sample data must be collected and then extrapolated over the entire population. This leads to a lack of variation and less accuracy in an estimate.

Next is trip distribution. This phase connects the predicted trips from one zone to destinations in other surrounding zones. Trips are decided upon based on attractiveness, distance, and travel time. The most popular trip destination method is called the gravity model. The attractiveness of a zone is proportional to its size but inversely proportional to the distance between origin and destination zones. One problem that can arise is trips originating from zones outside the study area and ending in or passing through the study area. This can throw off the equilibrium of the origin and destination zones, but can be rectified if the outside areas are included in the study. Also, it is a fixed model that must be recalibrated for significant changes in the transportation system.

The third step is modal split. It is determined thusly: "…the number of trips distributed between zonal pairs is allocated between auto and transit on the basis of relative travel times and costs between modes, and also, in some cases, on the basis of selected socioeconomic characteristics of the origin zone and land use characteristics of the destination zone" (Domencich & McFadden 1975). Modal split can be affected by changes in policy or the transportation system. However, this step often fails to account for all factors involved in door-to-door travel time such as different forms of transit.

Route assignment is the fourth and final step. It assigns car trips to roadways based on minimum travel time and roadway capacity. The results of this step can actually be fed back into the trip distribution step to try and equilibrate the entire process. However, the impacts of attempted model equilibration are hard to track because the trip generation step does not include information about travel time. This step usually does not account for out-of-vehicle travel time. An emerging traffic assignment technology is called dynamic traffic assignment or DTA. This process attempts to simulate the timewise variability of traveler path choice based upon the varying path times network due to traffic loading. DTA generally produces assigned link traffic volumes that are much closer to counted volumes than traditional static traffic assignment algorithms.

In summary, there are several problems with the four-step model. It is not very behavioral in that it is based upon observed travel behavior (the present situation) and cannot be adjusted for human behavioral changes. People's choice as to what time of day they choose to make trips is very rarely accounted for. Model equilibration is hard to do and therefore seldom seen. Lastly, the models are often based on a synthetic population generation and exact information cannot be acquired from everyone in the study area.

Chapter 3 – Capacity

In previous work done on this project, Eric Spurgeon (2008) goes into great detail concerning a study of the Highway Capacity Manual and the formulas used in the project. The following will briefly discuss these formulas and under what conditions they are to be applied.

3.1 HIGHWAY CAPACITY MANUAL

The Highway Capacity Manual (HCM) defines capacity as "the maximum hourly rate at which persons or vehicles reasonably can be expected to traverse a point or a uniform section of a lane or roadway during a given time period under prevailing roadway, traffic, and control conditions" (HCM, 2000). The capacity is a function of the physical properties of the roadway including, but not limited to, the number of lanes, shoulder widths, and lane widths.

Another important concept is that of Level-of-Service (LOS). The Highway Capacity Manual defines it as "a quality measure describing operational conditions within a traffic stream, generally in terms of such service measures as speed and travel time, freedom to maneuver, traffic interruptions, and comfort and convenience" (HCM, 2000). The six grades of LOS ranging from best to worst are A, B, C, D, E, and F. For controlled access freeways and certain other facility types, each of the grades A through E has a maximum flow rate known as the service flow rate. LOS F is representative of very poor and unstable flow and therefore has no definitive service flow rate.

Calculations to determine the capacity of a facility depend on the facility classification. The two types of facilities are uninterrupted and interrupted flow. Uninterrupted flow roadways (i.e., freeways) are completely access controlled and do not have traffic control devices. Interrupted flow roadways (i.e., arterials) have traffic control devices which have a direct effect on that facility's capacity. Interrupted flow calculations are somewhat more difficult as they involve knowledge of the location of traffic control devices and which roadways are meeting at specific intersections. Also, the presence of traffic control devices acts as a break in the road and may justify separating continuous segments into different links, each requiring its own analysis.

3.2 UNINTERRUPTED FLOW CALCULATIONS

3.2.1 Basic Freeways and Two-Lane Highways

The capacities of TxDOT's on-system interstates and freeways were calculated using the following formula from the 1985 HCM.

$$SF_i = MSF_i(N)(f_w)(f_{HV})(f_P) \qquad \text{Equation 3-1}$$

SF_i = service flow rate under prevailing traffic and condition for the level of service i (veh/h)

MSF_i = maximum service flow rate under ideal conditions for level of service i (veh/h)

N = number of lanes in one direction

f_w = adjustment factor for the effect of restricted lane widths and/or lateral clearance

f_{HV} = adjustment factor for the combined effect of trucks, buses, and recreational vehicles in the traffic stream

f_p = adjustment factor for the effect of driver population

The largest maximum service flow occurs at LOS E and is 2000 veh/h/lane at a speed of 60 mph or higher. The adjustment factor for lane width and lateral clearance (f_w) is taken from Table 7.2 of the 1985 HCM. The adjustment factor for the effect of driver population is 1.0 if the drivers are considered commuters, but this can be lowered if it is decided otherwise. The heavy vehicles factor is calculated using Equation 3-2.

$$f_{HV} = \frac{1}{1+P_T(E_T-1)+P_R(E_R-1)} \quad \text{Equation 3-2}$$

f_{HV} = heavy vehicle adjustment factor

P_T, P_R = proportion of trucks/buses and recreational vehicles in the traffic stream

E_T, E_R = passenger car equivalents for trucks/buses and RVs in the traffic stream

Values for the E_T and E_R variables can be found in Table 7.3 of the 1985 HCM. For the purposes of this project, the heavy vehicle factor (f_{HV}) for highways and freeways was calculated as 0.90 based on rolling terrain and the assumption of ten percent trucks making up the traffic (Spurgeon, 2008).

For two-lane highways, such as those with the US designation, the following equation was used from the 2000 HCM.

$$V = v_p(PHF)(f_G)(f_{HV}) \qquad \text{Equation 3-3}$$

V = demand volume for full peak hour (veh/h/ln)

v_p = passenger-car equivalent flow rate for peak 15-minute period (pc/h)

PHF = peak hour factor

f_G = grade adjustment factor

f_{HV} = heavy vehicle adjustment factor

The passenger car-equivalent flow rate for the peak 15-minute period (v_p) is 1700 pc/h as defined by the 2000 HCM. This is the maximum number of vehicles that can travel in one lane in one direction under base conditions. The peak hour factor (PHF) was chosen to be 0.95, under the assumption of rolling terrain. The grade factor (f_G) is 1.0 due to level or rolling terrain and moderate-to-high flow rates. The heavy vehicle factor (f_{HV}) was again 0.90 as was previously described.

3.3 INTERRUPTED FLOW CALCULATIONS

3.3.1 Urban Streets and Non-Access Controlled Highways

The point at which two or more roads meet at the same grade is subject to traffic control devices. Signalized intersections are the limiting factor in determining the capacity of urban streets and non-access controlled highways. When calculating capacity for the legs entering an intersection, each leg is called a lane group. "The HCM model does not produce a value for the capacity of the intersection. Rather, each lane group is considered separately, and a capacity for each is estimated" (Roess, et al., 2004). The reason for calculating the capacities of the individual legs is that traffic demand does not necessarily peak on all legs simultaneously. Green times must be apportioned to each leg respective of demand at a given time and signal timing can change over the course of a 24-hour period. Therefore, calculating a total intersection capacity is not particularly useful.

A lane group's capacity is calculated as a function of its saturation flow and its green time. "The saturation flow rate is the flow in vehicles per hour that can be accommodated by the lane group assuming that the green phase were displayed 100 percent of the time (i.e. g/C = 1.0)" (HCM, 2000). Saturation flow is calculated using Equation 3-5 from the 2000 HCM:

$$s = s_o N f_w f_{HV} f_G f_p f_{bb} f_a f_{LU} f_{LT} f_{RT} f_{Lpb} f_{Rpb} \quad \text{Equation 3-4}$$

s = saturation flow rate for subject lane group, expressed as a total for all lanes in lane group (veh/h)

s_o = base saturation flow rate per lane (pc/h/ln)

N = number of lanes in lane group

f_w = adjustment factor for lane width

f_{HV} = adjustment factor for heavy vehicles in traffic stream

f_G = adjustment factor for approach grade

f_p = adjustment factor for existence of a parking lane and parking activity adjacent to lane group

f_{bb} = adjustment factor for blocking effect of local buses that stop within intersection area

f_a = adjustment factor for area type

f_{LU} = adjustment factor for lane utilization

f_{LT} = adjustment factor for left turns in lane group

f_{RT} = adjustment factor for right turns in lane group

f_{Lpb} = pedestrian adjustment factor for left-turn movements

f_{Rpb} = pedestrian-bicycle adjustment factor for right-turn movements

The HCM recommended “default” ideal saturation flow of 1900 pc/h/ln is used as the base saturation flow, s_o. The various factors in the equation can be calculated from the formulas found in Exhibit 16-7 of the 2000 HCM. The most significant factors are those for left and right turns, f_{LT} and f_{RT} respectively. The right-turn factor is assumed to be 0.95 because it is the “average value of the right turn conditions discussed in HCM 2000 Exhibit 16-7” (Gao, 2008). The left-turn factor is assumed to be 0.95 if there is an exclusive left turn lane complete with protected phasing, as seen in Exhibit 16-7. However, if left turners are using a shared lane and do not have protected phasing, the factor would be assumed to be 0.90. When using 0.90, there is a risk of overestimating the intersection's capacity due to left turners queuing more than expected and holding up through traffic. The product of all other adjustment factors, $f_w f_{HV} f_g f_p f_{bb} f_a f_{LU} f_{Lpb} f_{Rpb}$, is assumed to be 0.95.

Following the calculation of saturation flow for the specified lane group, the capacity for that group is calculated using Equation 3-6 from the 2000 HCM.

$$c_i = s_i \left(\frac{g_i}{C}\right) \quad \text{Equation 3-5}$$

c_i = capacity of leg i (veh/h)

s_i = saturation flow of leg i (veh/h)

g_i /C = effective green ratio for leg i

The g_i/C ratio was calculated as a function of the volumes entering the intersection according to Equation 3-6:

$$\frac{g_i}{C} = \frac{(C - \sum_{i=1}^{n} Y_i)\frac{V_i}{\sum_{i=1}^{n} V_i}}{C} \qquad \text{Equation 3-6}$$

g_i = green time for leg i (sec)

C = cycle length (sec)

Y_i = clearance interval for phase i (sec)

V_i = volume on leg i (veh/h)

n = total number of legs at intersection

Due to the fact that the cycle lengths, green times, and clearance intervals are not given, the green time to cycle length ratio per leg is assumed to be proportional to the volume of that leg relative to the sum of all legs. Equation 3-6 is applicable to intersections with any number of legs. Clearance intervals were uniformly considered to be 4 seconds per phase. The cycle lengths are assumed to be 100 seconds for non-diamond intersections and 150 seconds for diamond interchanges along freeways.

A diamond intersection is actually two separate, but closely spaced, intersections coordinated to act as one. In Texas, they are generally found where freeway users are entering or exiting frontage roads running parallel to the freeway. These frontage roads act as arterials for access to driveways or access to intersections with crossing roads. There is generally heavy traffic flow, especially right and left turn movements, around these intersections due to proximity to the freeway. Due to the interdependency of traffic movements and the relative short spacing between the intersections, there are concerns about queuing. Queues have a detrimental impact on the effective length of a link, and this is primarily true for the link connecting the two intersections. To allow more green time to be apportioned and thus maximize flow, the cycle length is higher than normal intersections. Therefore, 150 seconds is used in lieu of 100 seconds.

As far as a signal timing plan is concerned, it is assumed that each leg has its own phase with a protected left turn movement, that is, a four green phase timing plan. This assumption may underestimate the capacity of the legs because signal timing plans are often customized to specific intersection traffic conditions at different times of the day. Also there is a multitude of concurrent, or non-concurrent, turning movements that can be chosen to maximize traffic flow, none of which can be rationally considered here.

3.3.2 Intersection Capacity Calculation Tool

A tool was developed within the GIS to aid in the capacity calculations of signalized intersections. This tool must be loaded into ArcView and has two buttons: CapacityIntersection and Google Earth Click. It allows the user to first select the legs of the specified intersection in the Demand network, as seen in Figure 3.1. Note that in this figure, the signals, represented by the orange dots, are offset from the roadways. This occurs because the signals geographically line up with the Census Roads, not the Demand, shapefile. The Demand network is used because it holds the AM and PM peak hourly volumes that are used in the calculations for intersection capacity. After the legs are selected, the user clicks on the “CapacityIntersection” button and a form (Figure 3.2) is opened which has preselected the necessary information from each road’s attribute table. This includes the road names and the AM and PM peak hourly volumes. The tool then sums up all the AM volumes and all the PM volumes separately to find which has a higher total and thus is the critical time period for that intersection. After calculating a saturation flow for each leg using Equation 3-4, the g/C ratio is calculated using Equation 3-6. Then by multiplying each leg’s saturation flow by its respective g/C ratio (Equation 3-5), the capacity for each leg is calculated. It is also important to note that the fields in the form are fully editable. If volumes need to be changed from the default values held in the Demand database, it can be done. Whether or not the cycle length should be considered 100s or 150s is controlled by checking or clearing a box next to "Diamond" in the Calculate form.

The other button is Google Earth Click. When this button is selected, the user can click on the ArcView map and a Google Earth window opens and zooms in on the location chosen. This proves very useful when trying to verify certain roadway layouts and lane configurations.

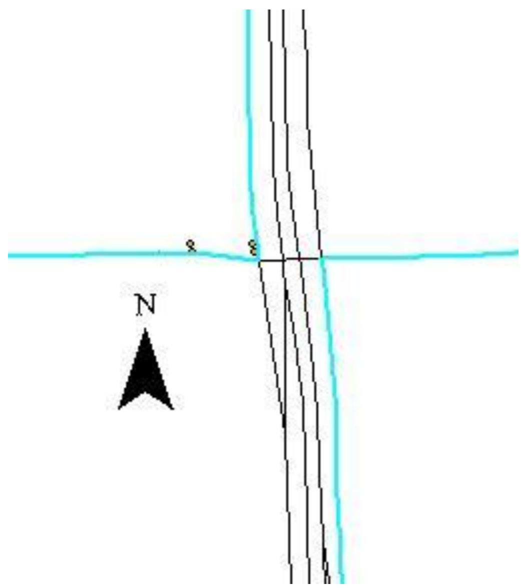


Figure 3.1 Selection of Intersection Legs

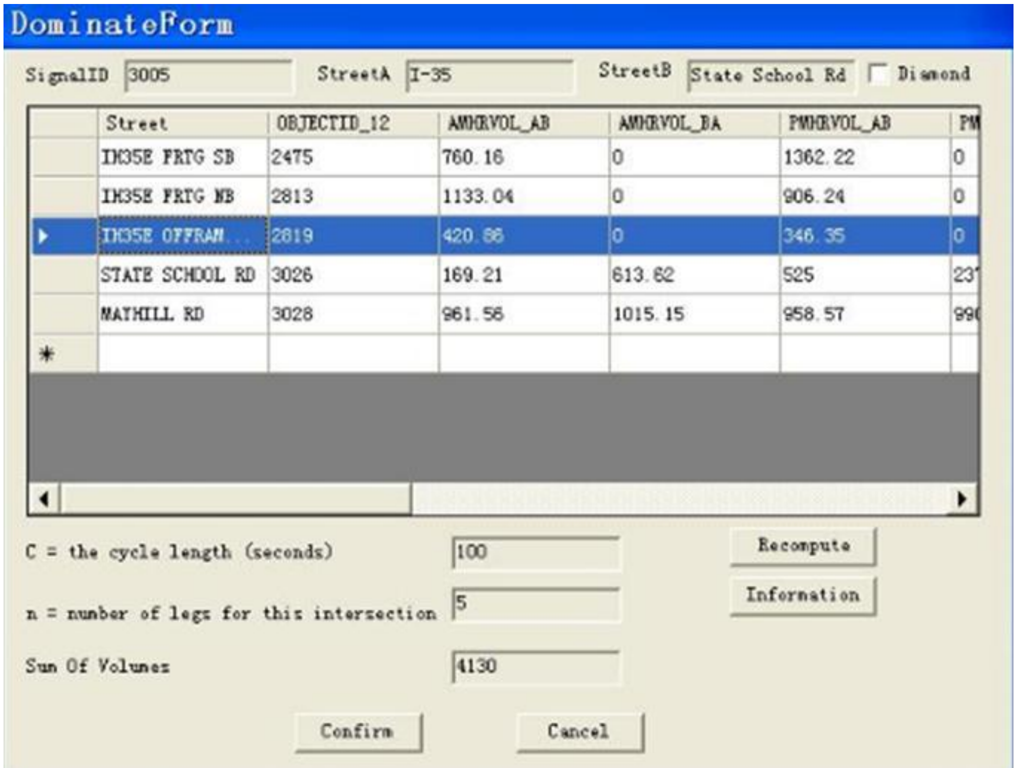


Figure 3.2 CapacityIntersection Form

3.4 CAPACITY DATA SOURCE

The roadway properties needed for capacity calculations include lane width, shoulder width, and number of lanes. This information is stored in a pavement performance monitoring database called the Pavement Management Information System (PMIS). TxDOT manages this database as a means of determining when certain sections of roadway are in need of repair. The PMIS stores data in discrete sections using the Texas Reference Marker (TRM) system. TRM is a statewide linear referencing database used for all on-system roadways in Texas (Zhang, et al., 1999). The PMIS sections are stored in half-mile increments which are tied to two TRM markers via two displacements.

The PMIS database has all the properties necessary to calculate roadway capacity and the TRM links this information to a specific geographical location. For this study, the PMIS data is stored in a Microsoft Access database while TRM information is stored as a collection of points in the GIS. The PMIS data is stored on a mainframe computer, but the information can be downloaded to a spreadsheet as in this project.

Unfortunately, the PMIS data is not always entirely accurate as some records may be outdated. If, for example, the number of lanes is misreported in the PMIS, it can always be checked with the Google Earth tool that has been developed.

3.5 SUMMARY

When calculating roadway capacity, a distinction is made between facilities with uninterrupted flow and those with interrupted flow. Basic freeways and some two-lane highways are examples of uninterrupted flow because they are completely access controlled and do not have traffic control devices. Urban streets and non-access controlled highways are examples of interrupted flow due to the presence of traffic control devices which limit roadway capacity. Equations from the Highway Capacity Manual are used throughout this project to calculate roadway capacity and a special tool has been developed to aid in the calculations for segments controlled by signalized intersections. The calculations are dependent upon having roadway characteristic information including number of lanes, lane width, and shoulder width. This information is taken from the PMIS database and linked to geographical points via the TRM system. The following chapter will discuss the other very important set of data, demand.

Chapter 4 – Demand

4.1 DEFINITION AND SOURCE

The number of people using the roadways (demand) is stored in a regional travel demand model. This information is reported as the number of vehicles passing a point on the roadway over the course of an hour. The peak hour is the sum of four consecutive 15-minute intervals that produce the highest demand. For this project, the AM and PM peak demands for every roadway will be used because they represent the critical values when considering traffic. The demand flows for the Dallas district were taken from the North Central Texas Council of Governments (NCTCOG) Transportation Department's Dallas - Fort Worth Regional Travel Model (DFWRTM). NCTCOG is a voluntary association of governments covering 16 counties centered around the Dallas – Fort Worth metroplex. The coverage area of this model is seen shaded in red in Figure 4.1. It was created to coordinate planning and avoid replication of work in the Central Texas region.

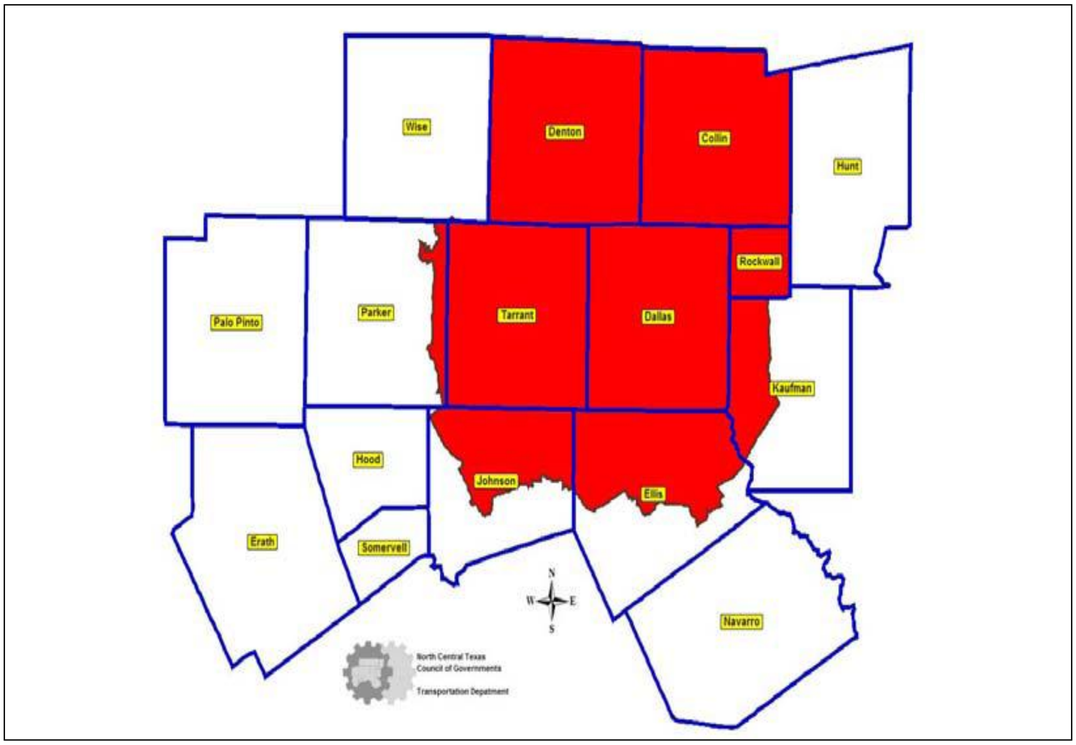


Figure 4.1 Dallas – Fort Worth Regional Travel Model Coverage Area
Source: Dallas - Fort Worth Regional Travel Model (DFWRTM): Model Description

The DFWRTM uses the aforementioned UTMS four-step model to predict traffic demands along roadways in its jurisdiction. It is run on TransCAD 4.8, which is another form of GIS software combined with transportation modeling features (Caliper, 2009). This model uses demographics, the road network, and mass transit systems as inputs and produces traffic demands and speeds on roadways while also estimating transit use. The vehicular traffic demand values used for this project were exported from TransCAD and are now stored in GIS shapefiles in vector format as a series of line segments representative of roadways. As was seen in Figure 1.1, this Demand layer does not properly align with the Census Roads layer in the GIS.

It should be noted that this model only produces expected flows, but it is the best estimate of traffic volume available. This is used in lieu of visiting each site and performing actual traffic counts. Such counts would be very labor intensive and time consuming, making them an unreasonable undertaking. If some of the Demand values do look unusual, it may be possible to check these numbers against actual traffic counts. TxDOT's intelligent transportation system, DalTrans, "utilizes Autoscope video detectors, inductive loops, and SmartSensor side-fire microwave detectors" to collect data in 5-minute increments (DalTrans, 2009). While these are not currently deployed at many locations, it is still often a reliable source. The data is available at:

http://ttidallas.tamu.edu/detectordataarchive/daltrans/default.htm

4.2 SUMMARY

In basic terms, demand is defined as the number of vehicles passing a point on a roadway during a one-hour period. In this project, the AM and PM peak hour demands are taken from the DFWRTM. This static model generates expected flows and these values are converted and stored in GIS shapefiles. While not as accurate as actual traffic counts, these estimates are the best available information and, in some instances, can be checked through a TxDOT database. The next chapter will discuss how demand, capacity, and the roadways' geographical locations are merged to form one database.

Chapter 5 – Merging the Databases

There is an 8-step process which is undertaken in order to create the desired aggregate GIS database. The steps involve creating a uniform identifier for all databases, assigning the start, mid, and end points of each roadway segment with coordinates, and then merging the databases together to get the final product.

5.1 CREATE PMIS LABELS FOR THE CENSUS ROADS LAYER

It was decided that the original PMIS naming convention, henceforth called "labels," would be used as the standard uniform identifier throughout the project and thus the identifiers for Census Roads and Demand will be changed to reflect this. First, load the Census Roads layer for a specific county in ArcView. Then select the roadway segments to be edited (Figure 5.1).

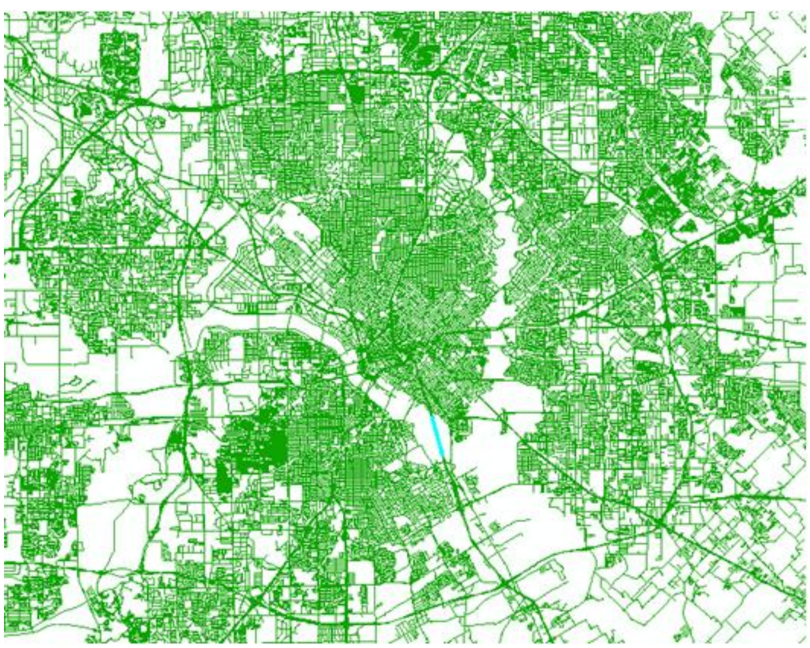


Figure 5.1 Census Roads Shapefile with Selected Segments Highlighted

Open the attribute table for Census Roads and click the “Selected” button to view the selected segments. Export the attribute table for the selected records to Microsoft Excel (Figure 5.2).

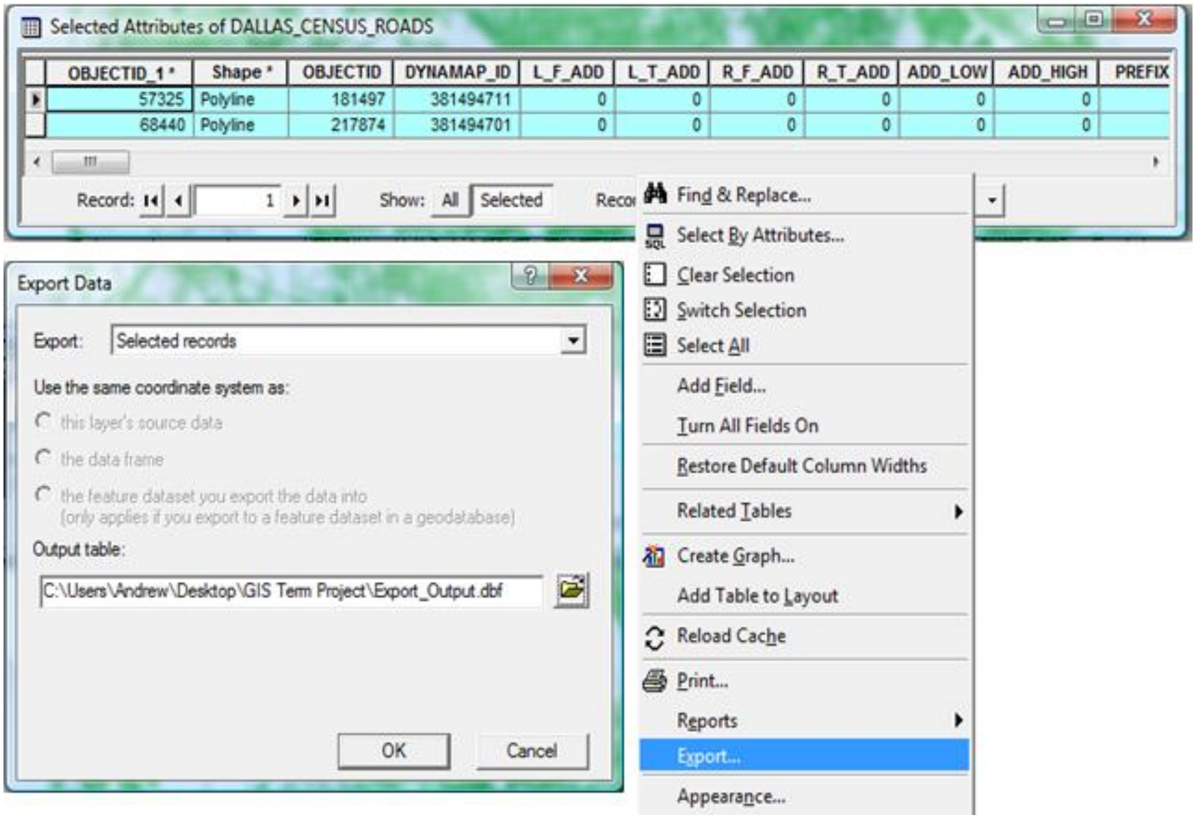


Figure 5.2 Export Selected Data

Once the selected records are in Excel, a macro (CENSUS_ROAD_ID_V1_5) is run to transform the Census Roads identifiers into PMIS labels (Figure 5.3).

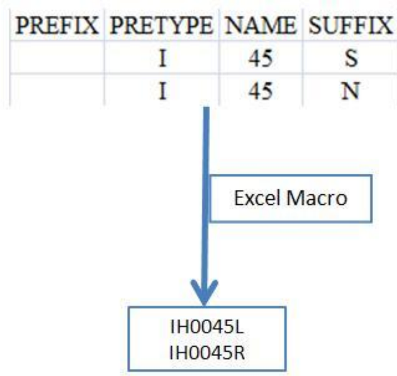


Figure 5.3 Before and After Roadway Labels

The example given in Figure 5.3 is for Interstate Highway 45 south and north. In this project’s naming scheme, L represents westbound or southbound while R represents eastbound or northbound. Once the new labels have been created, return to ArcView and join the newly created Excel table to the original attribute table (Figure 5.4).

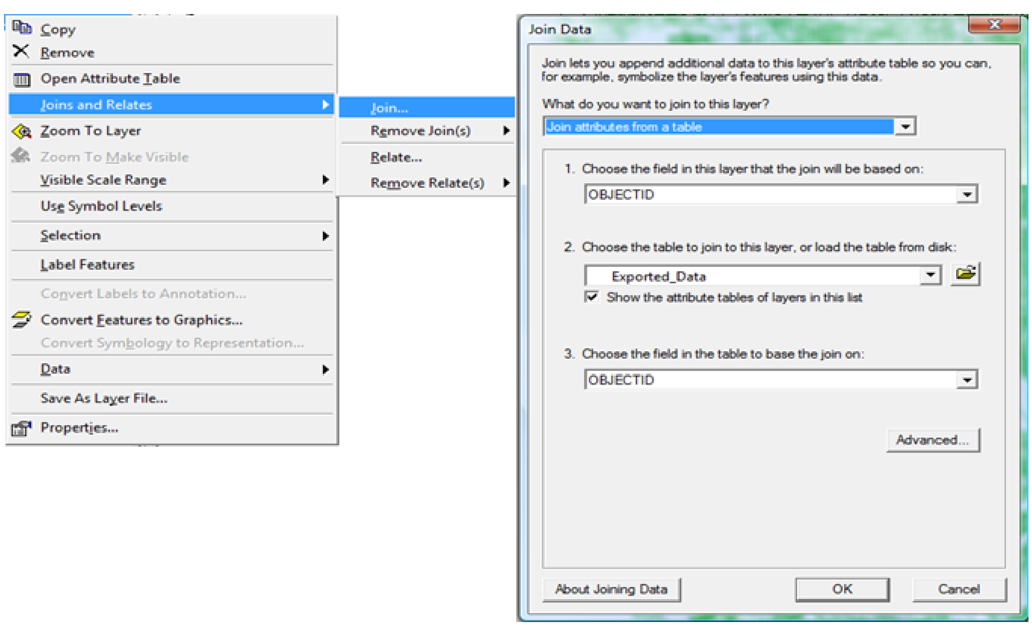


Figure 5.4 Joining Excel Table to Attribute Table

Once the join is complete, a new column has been added to the attribute table called HWY_ID_PMI. This contains the newly created identifier. However, the Census Roads attribute table should not be dependent upon a Join. Use the Add Field command to create a new column, in this case HWY_ID_P_1 (Figure 5.5).

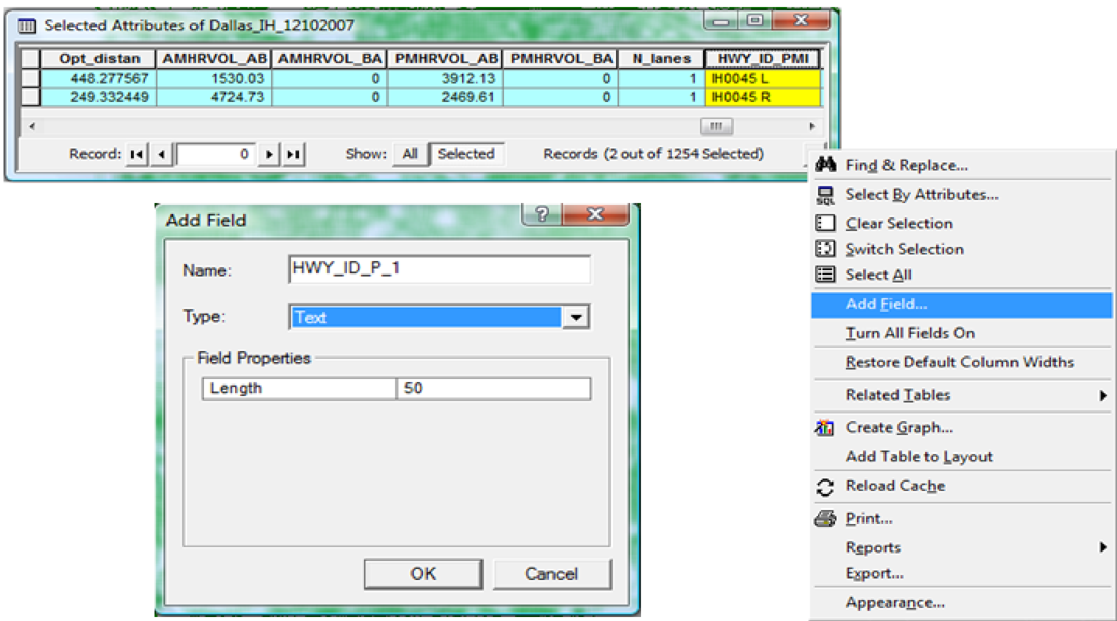


Figure 5.5 Adding a New Field

After the new column HWY_ID_P_1 has been created, use the Field Calculator to set the values in the new column equal to the values in the joined HWY_ID_PMI column (Figure 5.6).

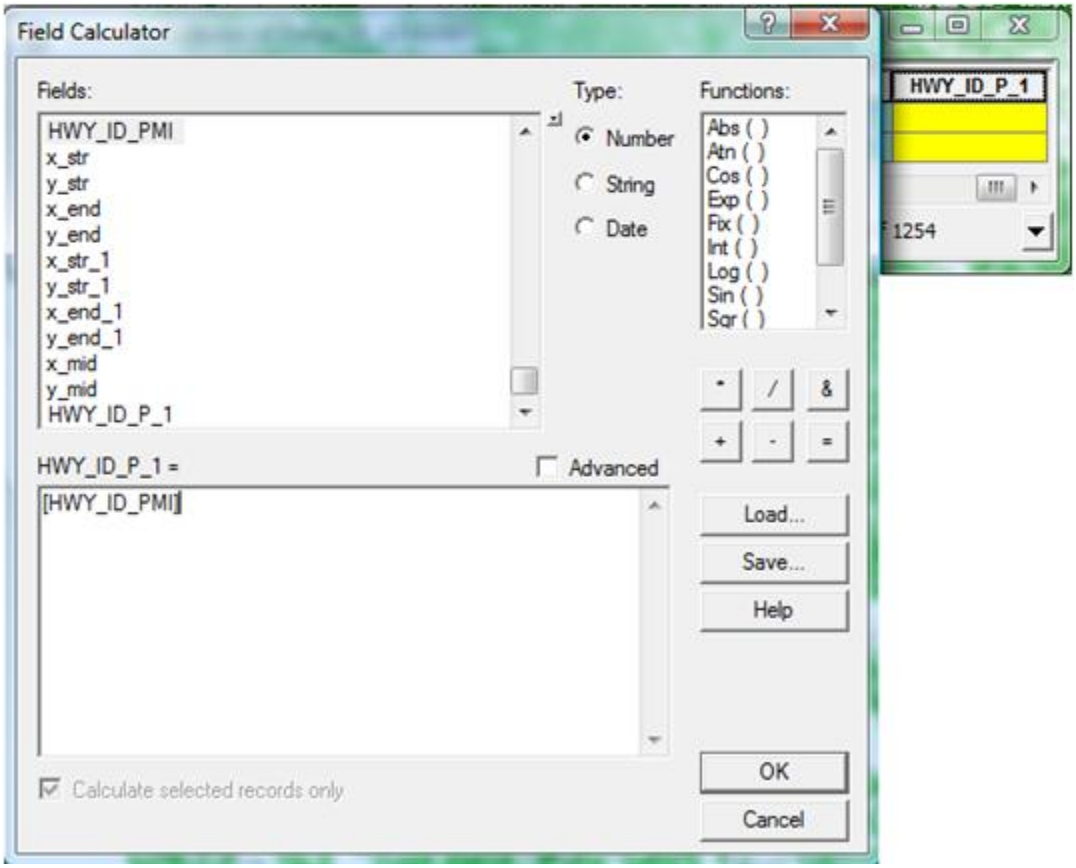


Figure 5.6 Using Field Calculator

After the values for the new column HWY_ID_P_1 have been filled, remove the Join (Figure 5.7).

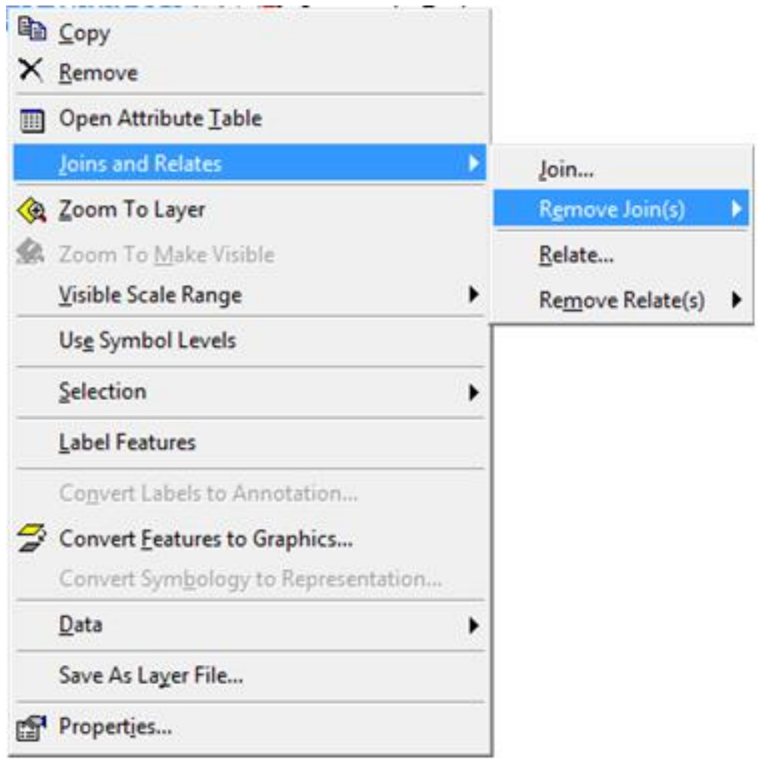


Figure 5.7 Remove Join(s)

5.2 ADD X & Y COORDINATES OF START, MID, AND END POINTS FOR CENSUS ROADS

Open the attribute table for Census Roads and go to the selected segments. Use the Add Field option to create 6 new columns for the coordinates (Figure 5.8). These columns will be labeled x_str, y_str, x_end, y_end, x_mid, and y_mid.

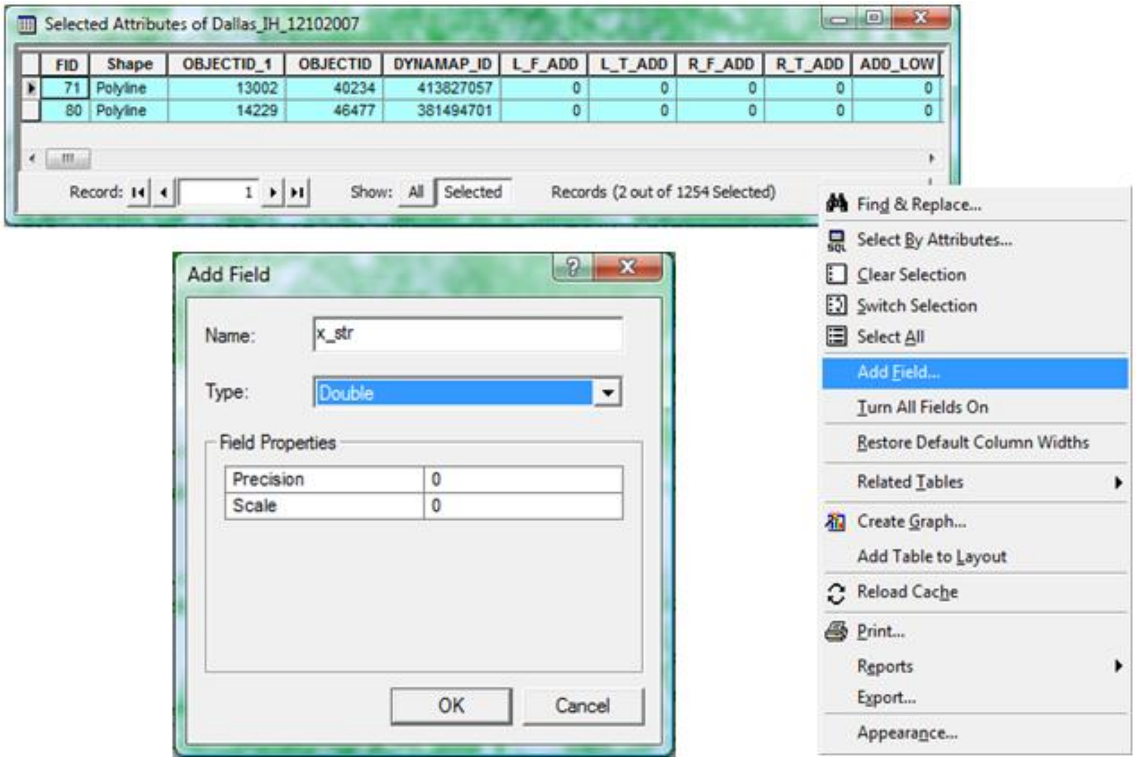


Figure 5.8 Use Add Field Tool to Create New Columns

Once the new columns have been created, use the Field Calculator to load the programs which calculate the coordinates for each respective column (Figure 5.9). Be sure to note that these programs are run by Field Calculator in ArcView, not as an Excel macro. The programs are named x_start_point, x_middle_point, x_end_point, y_start_point, y_middle_point, and y_end_point.

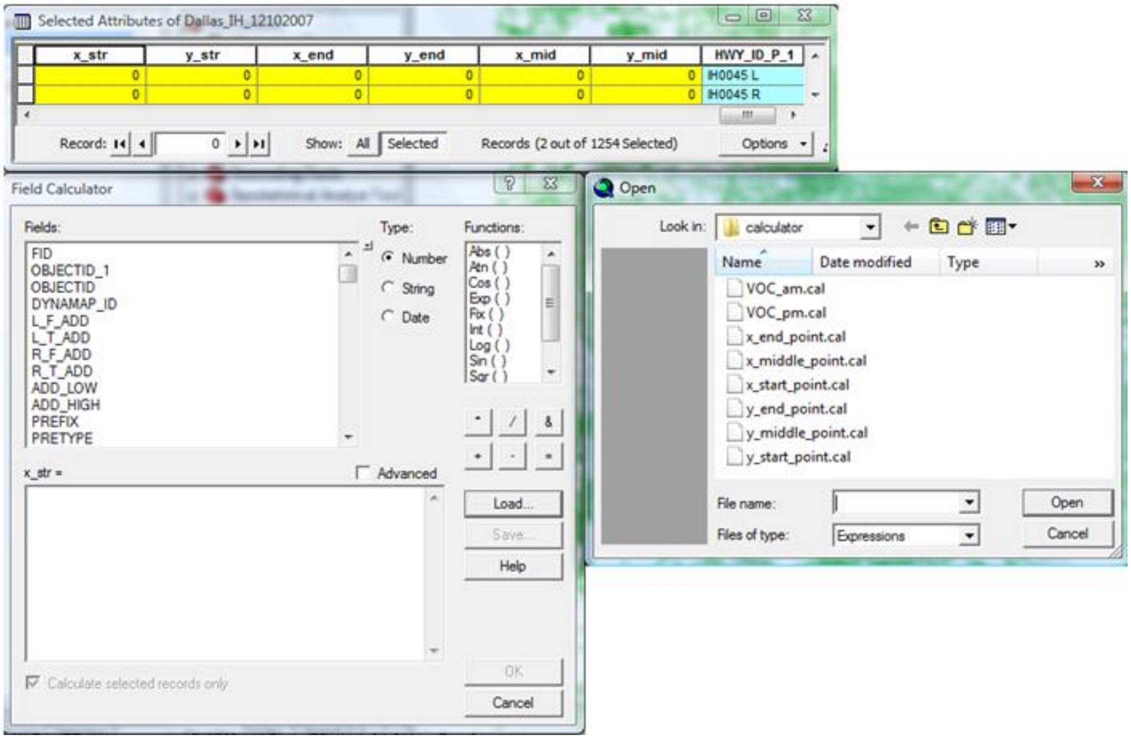


Figure 5.9 Use Field Calculator to Fill Values for Coordinate Columns

The columns have now been filled with their respective coordinates (Figure 5.10).

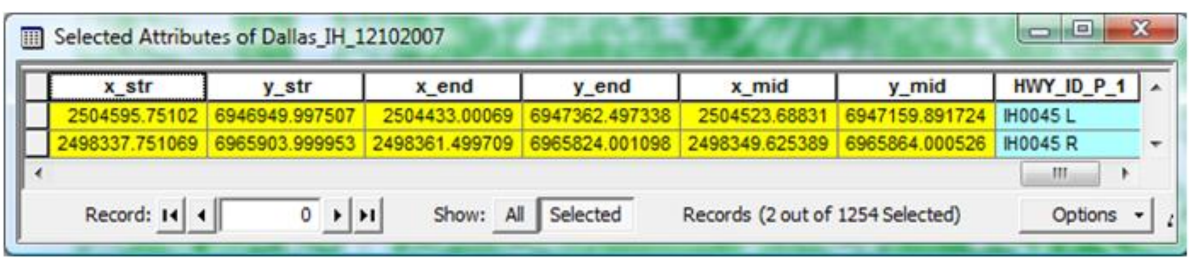


Figure 5.10 New Columns with Coordinates

5.3 CREATE PMIS LABELS FOR THE DEMAND LAYER

Load the Demand shapefile into ArcMap and repeat Step 1 for the Demand layer instead of the Census Roads. Use the macro CREATE_ID_4_DEMAND in place of the one used for the Census Roads.

5.4 ADD X & Y COORDINATES OF THE MID-POINTS OF THE DEMAND LAYERS

Repeat Step 2 using the Demand layer instead of the Census Roads layer. These coordinates will be used in the merging process in Step 5.

5.5 MERGE THE DEMAND AND CENSUS ROADS LAYERS

Export both attribute tables (Demand and Census Roads) to separate Excel sheets in the same workbook. A VBA program called MERGE_GIS_AND_DEMAND_V1_4 is run to merge the layers based on a uniform identifier and the coordinates calculated in previous steps. This step adds columns for the traffic volumes on each link to the Census Roads Excel sheet. Then join the Census Roads Excel sheet to the Census Roads attribute table in ArcGIS so that the AM and PM peak hourly volumes are now assigned to their proper roadway segments in the Census Roads layer (Figure 5.11).

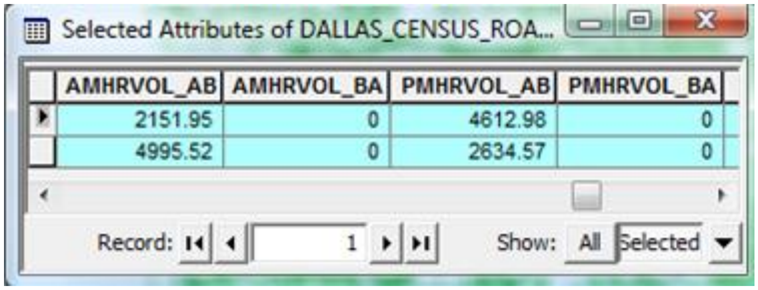


Figure 5.11 New Columns Added for AM and PM Peak Hourly Volumes

5.6 CALCULATE CAPACITY USING PMIS DATA

For this project, all PMIS data is stored in Microsoft Access. This database has much more information than is needed for this project. It is necessary to first query out the desired data and then export that data into Excel where it can be used to perform capacity calculations. The specific properties that were queried are seen in Table 5.1.

PMIS_DATA_COLLECTION SECTION	
PMIS Fields	**Purpose**
FISCAL_YEAR	used to select only the most up to date records
RESPONSIBLE_DISTRICT	PMIS exists throughout Texas, only interested in Dallas District
COUNTY_NBR	Dallas District consists of 7 counties
PMIS_HIGHWAY_SYSTEM	Two letter code that indentifies roadway type
SIGNED_HIGHWAY_RDBD_ID	Roadway code, name, and direction
BEG_REF_MARKER_NBR	TRM marker number, used to calculate DFO
BEG_REF_MARKER_DISP	Displacement in miles
END_REF_MARKER_NBR	TRM marker number, used to calculate DFO
END_REF_MARKER_DISP	Displacement in miles
NUMBER_THRU_LANES	Used to determine capacity
TOTL_SURF_RDWAY_WIDTH_MEAS	Divided by "NUMBER_THRU_LANES" for lane width
SHOULDER_WIDTH_RIGHT_MEAS	Used to determine capacity
SHOULDER_WIDTH_LEFT_MEAS	Used to determine capacity

Table 5.1 Queried PMIS Fields and Their Purpose
Source: Spurgeon, 2008

To calculate the capacities of freeway and highway links, Rui Gao (2008) wrote three Excel macros while Eric Spurgeon (2008) wrote one to calculate capacity. The process is as follows: While the PMIS data is stored in only half-mile increments, the actual properties of the roadways do not change at every individual section. Because of this, an Excel macro called PMIS_DATA_COMPILATION collapses records with the same highway ID, shoulder widths, and lane width. This is followed by another macro called DROP_LAST_LETTER_PMIS_ID. The TRM identifier is identical to the PMIS name until the last letter. Therefore, this macro removes the last letter from the PMIS identifier so the record can be matched to the TRM identifier. After this, a third macro is used to convert the TRM marker values to Distance from Origin (DFO) values while also matching the PMIS and TRM markers. It is called SEARCH_TO_MATCH_MARKERS. The DFO values are used to position the records properly in ArcView. In the case of freeways and US highways, a fourth macro is used called Capacity. As the name implies, it calculates the capacities of the links. For roadways controlled by signalized intersections, this macro is not necessary as the previously discussed intersection capacity tool calculates capacities of each leg of the intersection.

5.7 CREATE ROUTE EVENT LAYER OF CAPACITY DATA

Once the capacity for each roadway link is calculated, routes can be created using ArcView. First make sure that the Census Roads layer is added to ArcView but the Demand layer is not. Then select the desired roadway segments from the Census Roads shapefile. Once the segments are selected, make the selection into its own shapefile. Once this is done, go to the ArcToolbox, select Linear Referencing Tool, and choose Make Route Event Layer (Figure 5.12).

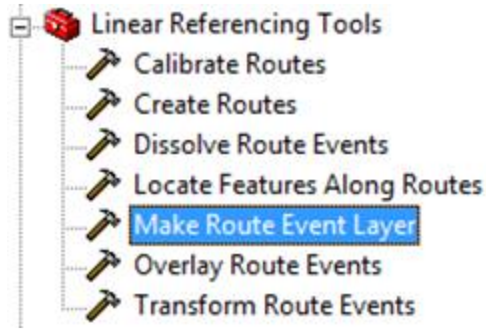


Figure 5.12 Make Route Event Layer Tool

Once this tool has been selected, the following form appears (Figure 5.13). This form is used to join the newly created roadway layer to the Excel sheet that was created in Step 6. This will create a new layer which assigns capacity values to the selected roadway. The PMIS database is only a set of numbers and it has no shapefile of road layout. It is linked to geographical positions via the Texas Reference Markers. Recall in Step 6 that the Distance from Origin was calculated based upon the TRMs. By using the Route Event tool, it is possible to assign these values to the appropriate locations in the ArcView display.

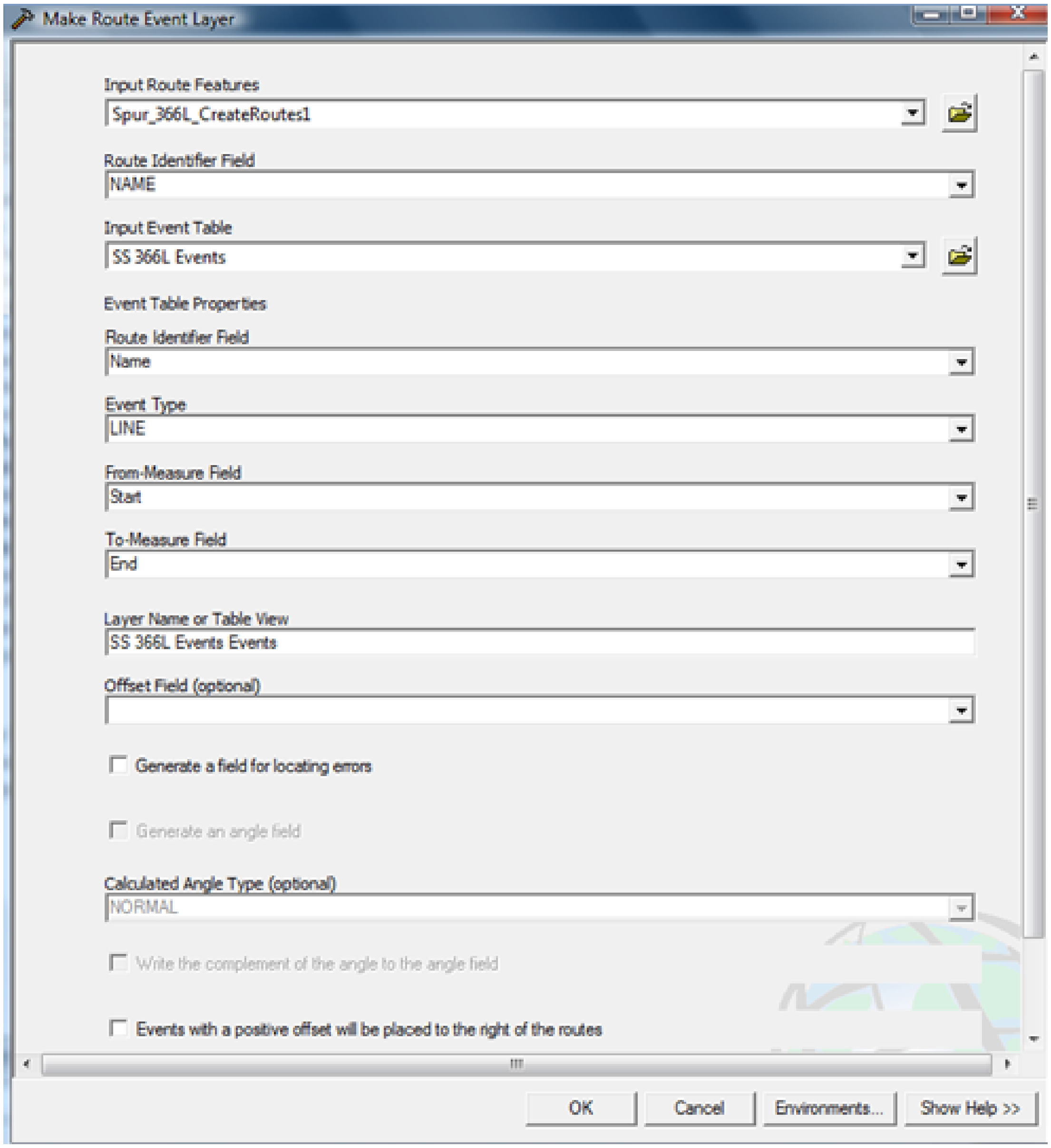


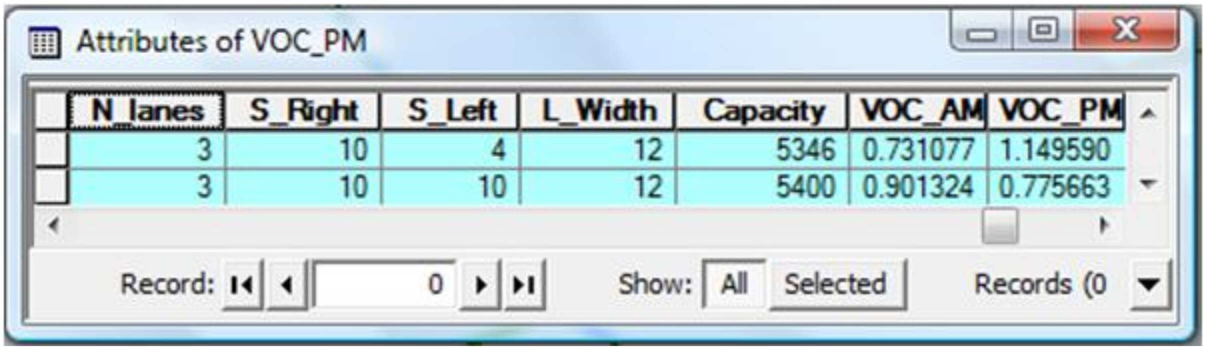
Figure 5.13 Make Route Event Layer Form

Once the form is completed, a Route Event layer is created which has taken many small segments and merged them to form a single stretch of road with uniform characteristics (Figure 5.14).

Figure 5.14 Completed Route Event Layer

5.8 MERGE THE CAPACITY (ROUTE EVENT) LAYER WITH THE CENSUS ROADS LAYER

This is accomplished the same way in which the Demand and Census Roads layers were merged in Step 5. Use the macro MERGE_GIS_AND_CAPACITY_V1_5 in place of what was run in Step 5. Merging these two layers provides the final additions to the Census Roads attribute table. The new columns are number of lanes (N_lanes), shoulder widths (S_Right and S_Left), lane width (L_Width), Capacity, VOC_AM, and VOC_PM (Figure 5.15). Note that VOC stands for Volume Over Capacity.



N_lanes	S_Right	S_Left	L_Width	Capacity	VOC_AM	VOC_PM
3	10	4	12	5346	0.731077	1.149590
3	10	10	12	5400	0.901324	0.775663

Figure 5.15 New Columns Added for Roadway Properties, Capacity, and VOC Ratios

5.9 SUMMARY

The separate databases used to create the GIS tool lack a common identifier and do not align spatially. Therefore, an 8-step procedure is employed to merge all the necessary information:

1. Create PMIS labels for the Census Roads layer.
2. Add x and y coordinates of start, mid, and end points for Census Roads.
3. Create PMIS labels for the Demand layer,
4. Add x and y coordinates of the mid-points of the Demand layers.
5. Merge the Demand and Census Roads layers.
6. Calculate capacity using PMIS data.
7. Create route event layer of capacity data.
8. Merge the capacity (route event) layer with the Census Roads layer.

Steps 1 and 3 fix the issue of not having a uniform identifier, or label, across all three databases. Steps 2, 4, and 5 solve the problem of mismatched layers due to the use of different coordinate systems. Steps 6, 7, and 8 assign the data from the PMIS database to the final GIS layer. Once this process has been completed, the capacity and demand will be merged in the same database and will be placed in their correct geographical location. This allows users to access the information via an interface and tool explained in the next chapter.

Chapter 6 – Current GIS Tool

6.1 TOOL AND INTERFACE

As of today, the access controlled on-system roadways throughout the Dallas district have been completed. This can be seen below in Figure 6.1.

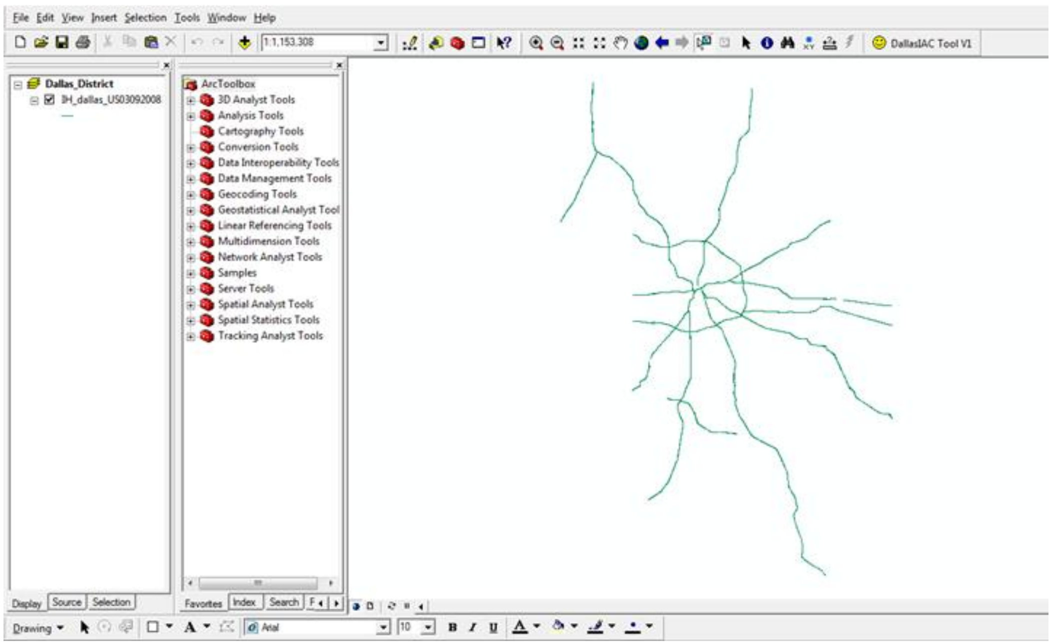


Figure 6.1 Interface for Access Controlled Roadways

With all the data now merged into one attribute table, a tool was created to access what a traffic engineer would be most interested in. This is the DallasIAC Tool V1 in the upper right-hand corner of the ArcView interface next to the smiley-face icon. The user must first load the final map into ArcView, then select the roadway segment to be studied and click this button. A pop-up form will be displayed as seen in Figure 6.2. This form displays many of the roadway characteristics including route name, direction of travel, number of lanes, lane width, shoulder width, capacity, and AM and PM peak volumes.

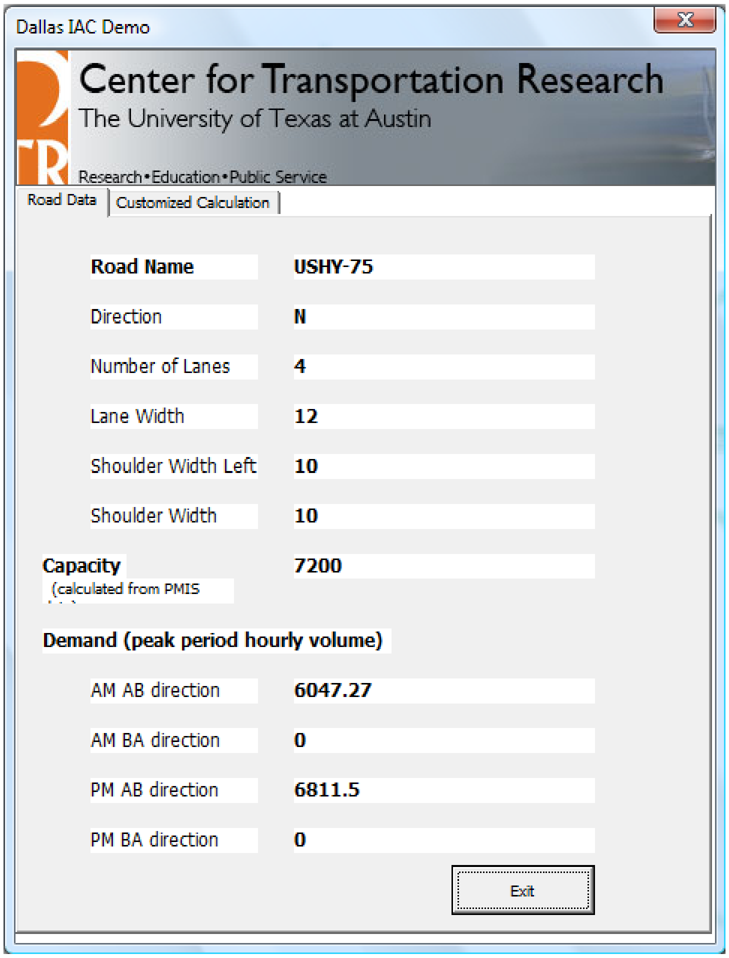


Figure 6.2 Pop-up Display

Towards the top of the tool is a tab for Customized Calculation. When this is selected it enables the user to edit several of the roadway characteristics: number of lanes, lane width, and shoulder width (Figure 6.3). By pressing "Recalculate" the new capacity will appear. This is an important feature as it allows a traffic engineer to modify these features, in the case of an accident or construction, and then see how this would affect the roadway capacity. Traffic detouring and planning can then be done accordingly.

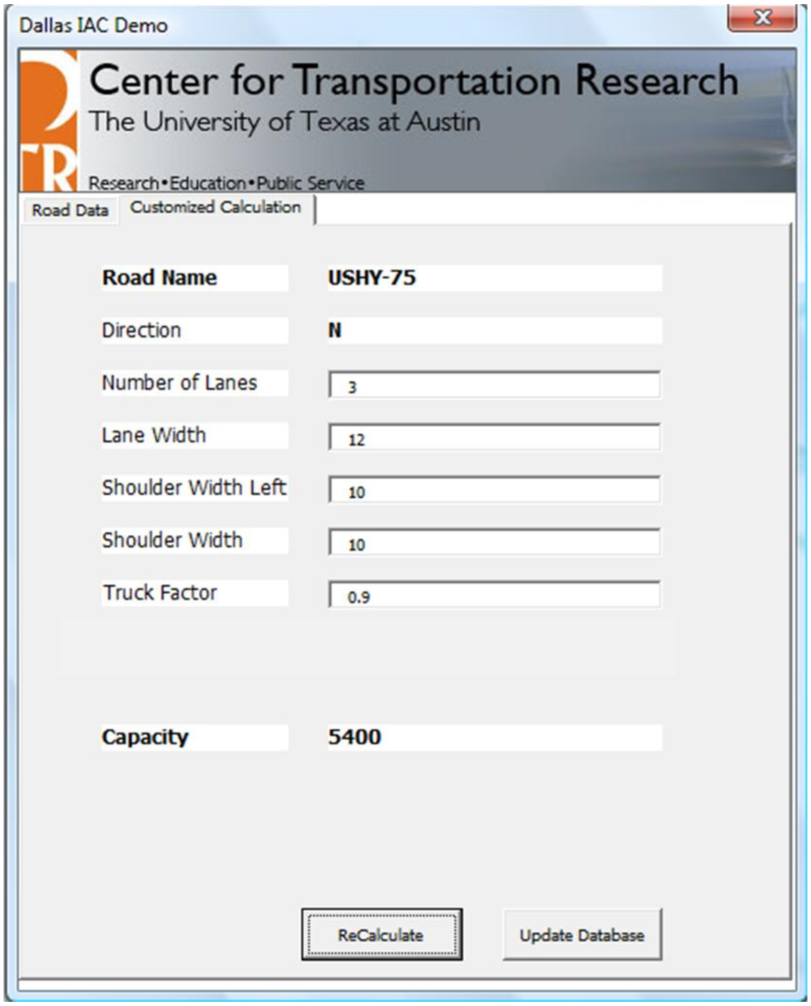


Figure 6.3 Customized Calculation Option

6.2 GOOGLE EARTH & GOOGLE MAPS FEATURES

In order to enhance the overall experience provided by this GIS model, two useful features have been added. These are Google Earth (Figure 6.4) and Google Maps (Figure 6.5). These tools allow the user to select a point in the map and then a pop-up window will appear which takes them to that point in Earth or Maps. The Earth option is very useful because it gives the user a real-world picture of the situation at the chosen point.

For the developers, it becomes especially useful if there is something unusual about the data. One such problem is that the PMIS data occasionally reports an erroneous number of lanes for a chosen segment of roadway. By checking the satellite imagery from Google Earth, the user can see how many lanes are actually at that point and adjust accordingly.

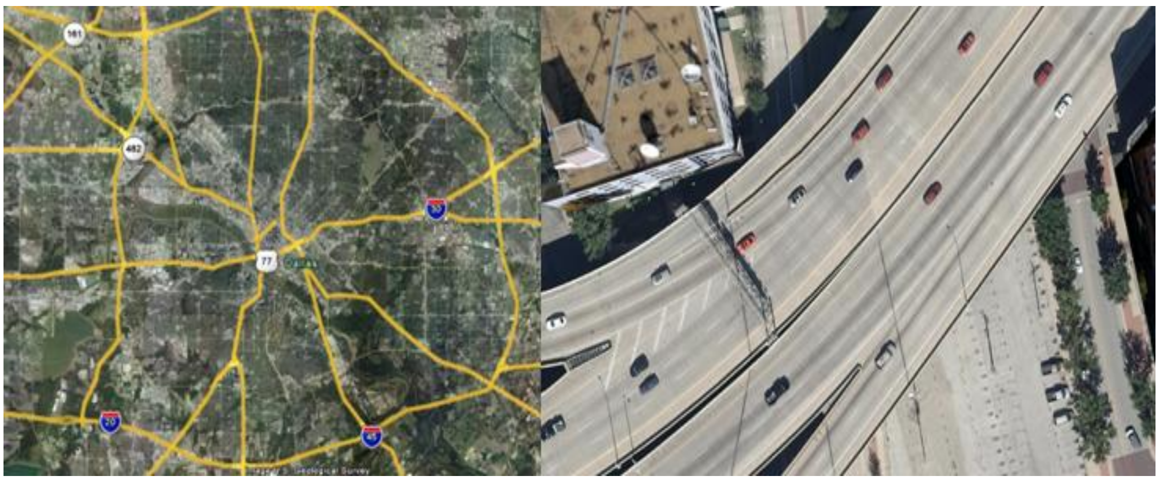


Figure 6.4 Google Earth Feature

The Google Maps feature is also interesting as it provides the viewer with an idea of congestion problems on a selected state or interstate highway. In Maps, the viewer can change the day and time and see if traffic is good, mediocre, or poor. This component can be very valuable when attempting to communicate a traffic situation to those in different professions. Transportation engineers must be able to effectively convey their findings and ideas to people such as lawmakers. Using something that most people are familiar with (i.e., Google Maps) can be a good common ground starting point.

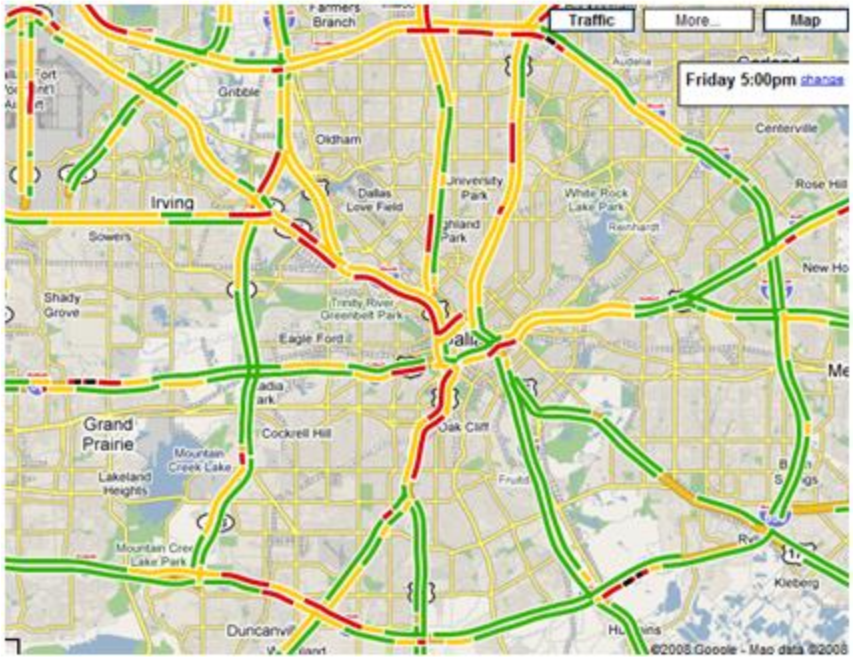


Figure 6.5 Google Maps Feature with Traffic Option

6.3 MAINTAINING GIS DATA ON AN INTERNET MAP SERVER

One last feature that is being worked on in coordination with the original GIS tool is an ArcIMS interface. When using ArcView, the data is only accessible on a computer that is equipped with both GIS software and the files used to create the whole interface. All this information can be sent to those who need to use it (i.e., TxDOT engineers), but whenever there is an update, it would all need to be re-sent. The idea of using ArcIMS allows the creator(s) to upload all data to an internet map server where it can be accessed and used by other engineers or anyone who has an interest in the project. It also makes accessing updated information very straightforward. This feature is currently up-to-date with all access controlled roadways available. The background is a non-clickable land use map.

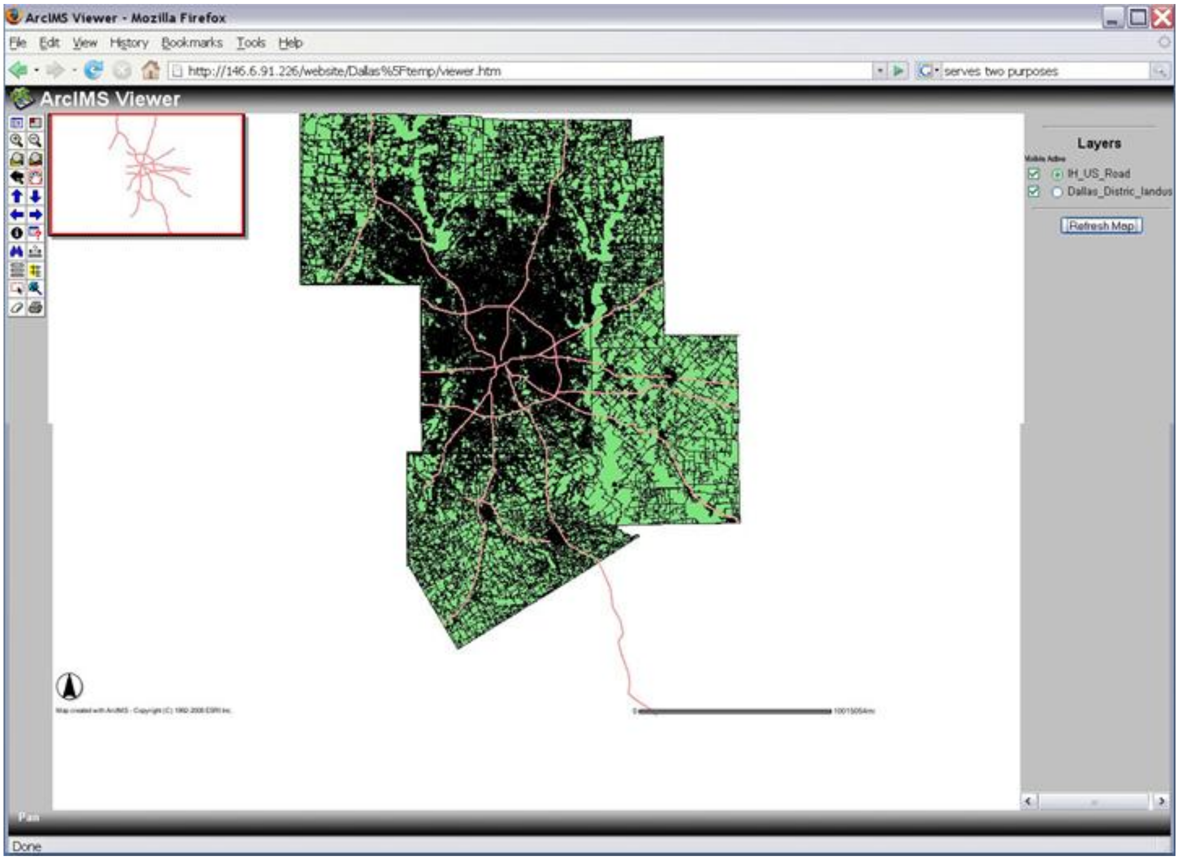


Figure 6.6: ArcIMS Interface

6.4 SUMMARY

After the separate databases were merged, a tool was created to provide access to a roadway's capacity, demand, and physical characteristics. The DallasIAC Tool V1 displays all of this and then allows for customization of certain roadway attributes to see how capacity would be affected. Other features include access to Google Earth and Google Maps for a selected location in the GIS. The ArcIMS interface will allow outside users to view, and eventually edit, the GIS. The next, and final, chapter summarizes the project thus far and discusses issues that will be dealt with in the future.

Chapter 7 – Summary

This GIS model is intended for use by traffic engineers in the Dallas district of TxDOT. In conjunction with the MUTCD, it will aid engineers in setting up traffic control plans (TCPs) to most effectively re-route traffic in the event of roadway construction or closure due to other reasons. The information about capacity, demand, and roadway layout is of absolute necessity when undertaking such planning. The static, clickable interface and user-friendly tools make it a simple way of finding the required information.

The geographical locations of roadways were taken from the Census Roads database. Capacity was calculated using roadway properties queried from the PMIS and linked to geography via the TRM system. Demand numbers were received from NCTCOG's DFWRTM which uses TransCAD and the four-step traffic forecasting model. The layers did not align properly by themselves, but by setting a common identifier and using a series of programs, an aggregate database was formed.

Perhaps the most important pieces of information about the roadways are the calculated Volume Over Capacity (VOC) ratios. These are representative of how congested a roadway is. Theoretically, this number cannot exceed 1.0 because traffic flow cannot exceed the capacity. For this project, the VOC ratio was allowed to go up to 1.2. While this was not a common occurrence, the allowance was due to the fact that there is some uncertainty with the Demand data being used. The Demand is the product of an estimation model, not real traffic counts. If the VOC exceeded 1.2, that section of roadway was looked at more closely via Google Earth and it was often found that the number of lanes was underreported by the PMIS database.

Currently, the tool contains information for TxDOT's on-system interstate highways and freeways in the Dallas district. The model is in the process of being updated to include frontage roads, arterials, and other major non-access controlled roadways. The first step towards this has been taken with the development of the signalized intersection capacity tool. As progress is made on these signal-controlled roads, several issues will need to be addressed.

The first of these issues is the naming convention. Since the freeways and interstates have been completed, the exact Excel macros that have been originally used will need some modification. Instead of dealing with IH35 north or south, the work now involves regular roadway names such as West Parmer Lane. If there is a case where a road may have two separate names (i.e., West Parmer Lane or FM 734) the databases will need to be checked for which name is used to refer to that section of roadway.

Another desire is to modify the pop-up window to be able to handle more than just one segment. Instead of only being able to view one short section of road at a time, an engineer will most likely need to see the information over a stretch of several miles.

The programs written by Gao were customized to work with the links for interstates and freeways. However, the links for frontage roads and arterials are often shorter than those. The programs will need to be tested for accuracy when used on these shorter links.

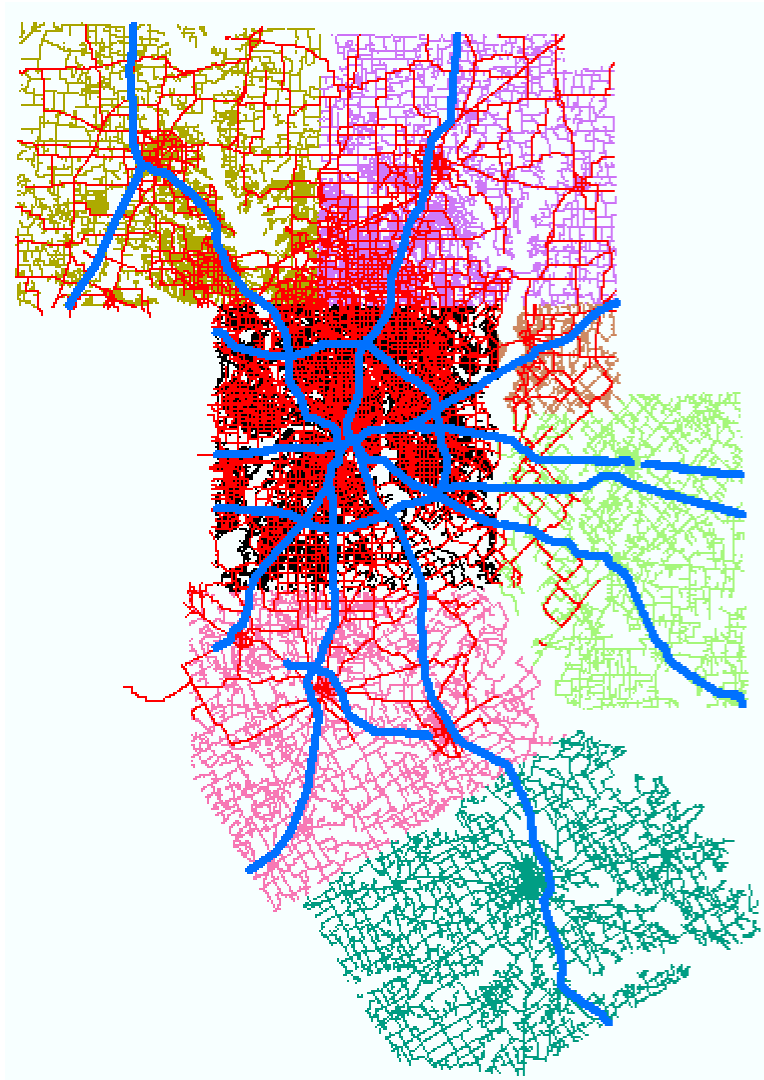
Lastly, another issue with the non-access controlled roadways is that in some instances a single roadway may be represented by one line in Census Roads, but represented by two lines in the Demand database. The cause for this often seems to be a median separating the north-south or east-west lanes. Extra caution will have to be taken when assigning Demand values to the Census Roads in these instances.

It is worth noting that the NCTCOG DFWRTM is a static model. This means that it produces demand values based on peak periods of the day. While these numbers are useful in that they produce the highest VOC ratios, this also misrepresents the entire day. Peak traffic does not last 24 hours. What is needed is a dynamic traffic assignment model which fluctuates demand throughout the day. This way, when a traffic engineer uses this tool in the future, there will ideally be another option to adjust for time of day. If the time periods for lane or shoulder closures are not fixed, common sense would dictate closures during typical off-peak periods. But a dynamic model gives engineers more freedom in determining how much of a road can be taken out of service during or between AM and PM peaks or overnight. When time becomes a constraint, it is best to work with dynamically-predicted traffic demand.

In coordination with the main tool development in ArcView, an ArcIMS interface is also being created. This will make accessing the tool much quicker and easier via the internet. By setting up an internet map server, a traffic engineer in Dallas does not need to have an ArcView license and all the specific files to view the maps. It will all be readily available online.

Appendix

Figure A.1 All Layers



Key: Blue = Interstates/Freeways
Red = Demand Network
Other = Census Roads

Figure A.2 All Layers with Signalized Intersections

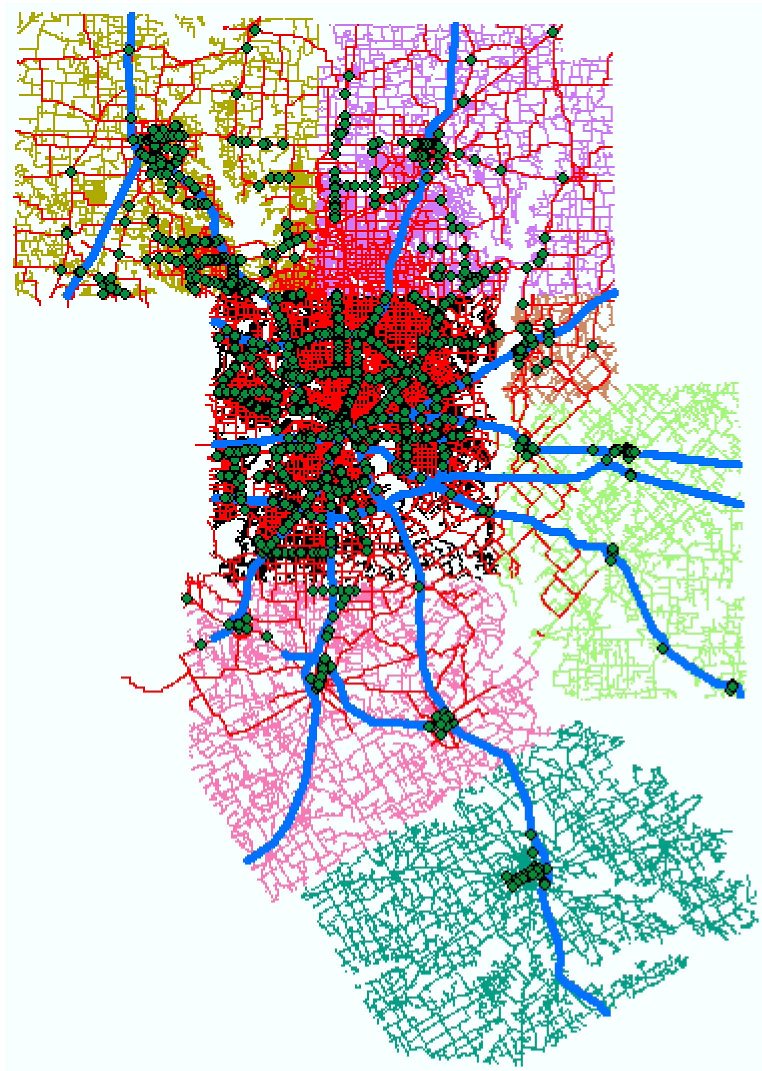
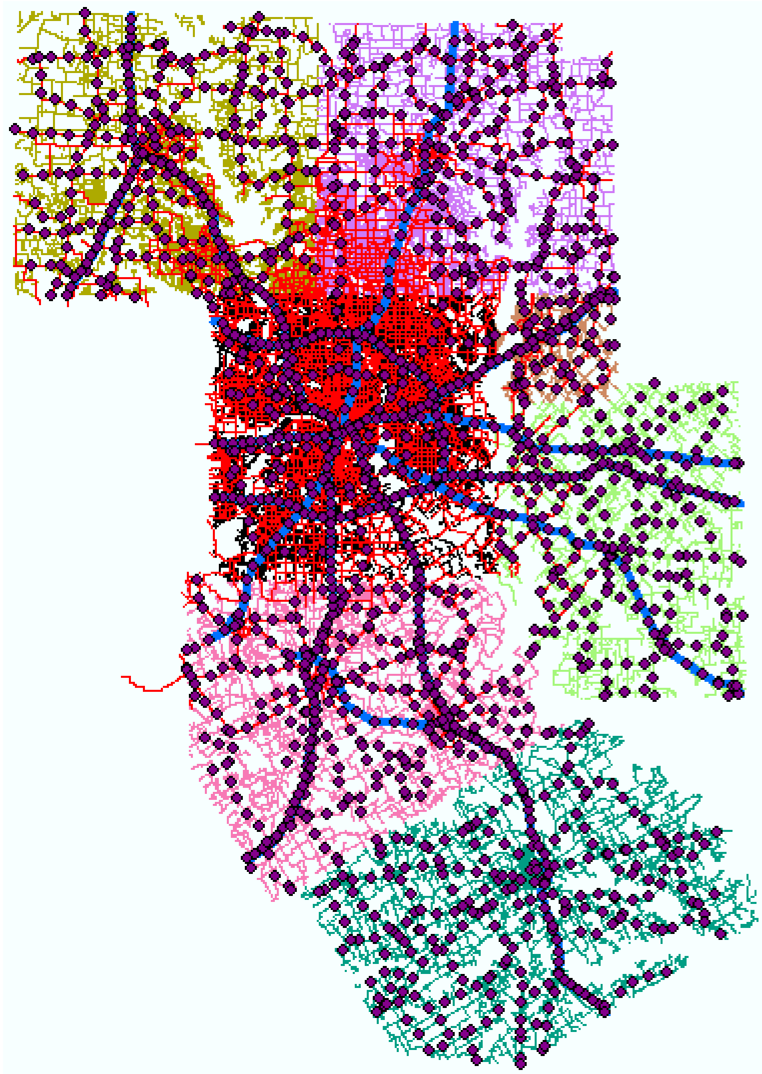


Figure A.3 All Layers with TRMs



Instructions A.1 Specific Commands for Merging Procedure (Example)

1. CREATE PMIS LABELS FOR GIS LAYERS
- master file -> demo -> GIS_Dallas_IH -> Dallas_IH_12102007.shp
- open attribute table
- options -> export
- save into new folder under Master File (Exported Data)
- do not add new table to current map
- open new file in EXCEL
- make new column after SUFFIX to join prefix and suffix [=IF(J2="",M2,J2)] , copy and paste over self as values, delete original prefix and suffix columns
- alt+F11 to load program
- file -> import file -> master_file -> Rui -> programs -> Census_road_ID_V1.5
- modules -> open program
- each column goes in # order, check that #'s over columns match those in program
- delete row of column #'s
- run -> run program
- say yes to everything
- 2 new columns: TRM & PMIS (delete TRM)
- copy and paste after Shape_Leng column, delete original PMIS column
- copy and paste entire sheet onto a new sheet
- delete everything on new sheet except OBJECTID & new column HIWAY_ID_PMIS
- save in 97-03 format
- go back to ArcMap, close attribute table, right click layer, go to Joins and Relates -> Join
- join attributes from table
1. OBJECTID
2. Open folder and browse to newest sheet
3. OBJECTID
- open attribute table again, new columns at end
- options -> add field (may need to save ArcMap, close, and reopen)
- Name: HWY_ID_P_1
- Type: Text
- right click new column -> field calculator -> Sheet1$.HIWAY_ID_PMIS
- right click layer -> remove Join(s) -> remove all joins

2. ADD X & Y COORDINATES OF START, MID, AND END POINTS OF GIS LAYERS (CENSUS ROADS)
- options -> add field -> Name: x_str, y_str, x_end, y_end, x_mid, y_mid / Type: Double
- right click each column -> erase whatever may be default value -> load -> C:/ -> Rui -> calculator -> x_start, y_start, x_end, y_end, x_mid, y_mid

3. & 4. CREATE PMIS LABELS FOR DEMAND LAYERS
ADD X & Y COORDINATES OF MID-POINTS OF DEMAND LAYERS (STEP 2)

- close attribute table
- add layer -> c:/ -> master file -> demo -> selected_demand -> export_output
- check layer's attribute table for PMI, mid_x, mid_y at end of table
- rename it Dallas_IH_Demand
- export attribute table, convert to an EXCEL file
- also export other GIS layer attribute table to EXCEL

5. MERGE DEMAND AND GIS LAYERS

- open EXCEL file
- rename sheet "Demand" then add new sheet "GIS"
- copy and paste GIS EXCEL sheet onto new "GIS" sheet
- file -> import file -> master_file -> Rui -> programs -> run program -> Merge_GIS_and_Demand_V1_4.bas
- check to make sure columns line up with program (can change program if needed)
- delete row of column #'s
- copy "GIS" sheet onto new sheet
- delete everything except OBJECTID, Demand_ID
- go to layer Dallas_IH_12102007 -> joins and relates -> joins

1. OBJECTID
2. whatever the new sheet is
3. OBJECTID

- open attribute table
- Dallas_IH_12102007_Demand_ID (may have to add this column)
- field calculator
- select [Sheet2$.Demand_ID]
- joins and relates -> joins -> remove all joins
- left click Dallas_IH_12102007 -> join data

1. Demand_ID
2. ◊ Dallas_IH_dem (should be in place already)
3. OBJECTID

- columns: Dallas_IH_12102007.AMHRVOL_AB, _BA, .PMHRVOL_AB, _BA)may need to add)
- fill these columns: Field Calculator -> [Export_Output.AMHRVOL_AB]
- do for all 4 columns

6. CALCULATE PMIS DATA USING CAPACITY
See: Creating a Route Event in ArcGIS Using PMIS Data in this Appendix

7. CREATE ROUTE EVENT LAYER OF CAPACITY DATA
See: Creating a Route Event in ArcGIS Using PMIS Data in this Appendix

8. MERGE CAPACITY (ROUTE EVENT LAYER) & GIS LAYER
- remove Demand layer
- add layer -> demo -> route_event -> add .shp files
- click red button (show/hide ArcToolbox)
- tools -> extensions -> enable all of them
- ArcToolbox -> data management tools -> general -> merge
- then input database, input the .shp files, give name and merge them
- remove the originals
- open attribute table
- ID column (may need to add) -> field calculator -> [FID] + 1000
- options -> add field -> mid_x & mid_y (may be there already)
- field calculator -> load -> calculator -> mid_x & mid_y
- follow Step #5 but instead of “Demand” use “Capacity”
...
1. ID
2. name of new sheet
3. ID
…
- in place of AMHRVOL_AB use: N_LANES, S_RIGHT, S_LEFT, L_WIDTH, CAPACITY

Creating a Route Event in ArcGIS using PMIS data

Formatting the PMIS Data in EXCEL

1. Query the Data from the PMIS Access Database table "PMIS_Data_Collection_Section"

Table 1: Column Headings to be used in Querying Data

PMIS_DATA_COLLECTION SECTION	
Column Heading	**Purpose**
FISCAL_YEAR	used to select only the most up to date records
RESPONSIBLE_DISTRICT	PMIS exists throughout Texas, only interested in Dallas District
COUNTY_NBR	Dallas District consists of 7 counties
PMIS_HIGHWAY_SYSTEM	Two letter code that indentifies roadway type
SIGNED_HIGHWAY_RDBD_ID	Roadway code, name, and direction
BEG_REF_MARKER_NBR	TRM marker number ***use table look up for DFO**
BEG_REF_MARKER_DISP	Displacement in miles
END_REF_MARKER_NBR	TRM marker number ***use table look up for DFO**
END_REF_MARKER_DISP	Displacement in miles

NUMBER_THRU_LANES	Used to determine capacity
TOTL_SURF_RDWAY_WIDTH_MEAS	Divided by "NUMBER_THRU_LANES" for lane width
SHOULDER_WIDTH_RIGHT_MEAS	Used to determine capacity
SHOULDER_WIDTH_LEFT_MEAS	Used to determine capacity

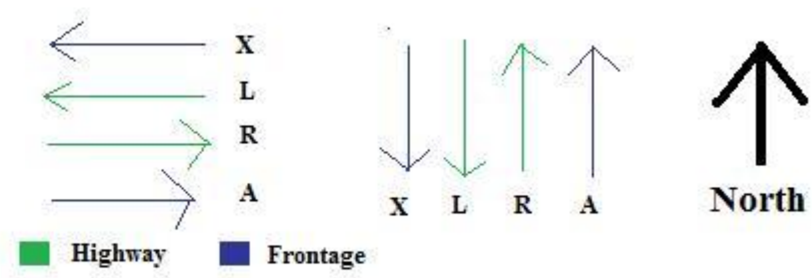


Figure 1: SIGNED_HIGHWAY_RDBD_ID Diagrams

The “RESPONSIBLE_DISTRICT” is Dallas, which has a numerical value of 18. The “COUNTY_NBR” values can be found in the PMIS table “TACS_Table_CNYCLMT”

Table 2: County Codes

County	Number
Collin	043
Dallas	057
Denton	061
Ellis	071
Kaufman	130
Navarro	175
Rockwall	199

2. Export the Query to Excel
3. Format Data in preparation of running macro “PMIS_Data_Compilation”
    a. Sort by SIGNED_HIGHWAY_RDBD_ID
    b. Create a new column “END” that is the sum of “END_REF_MARKER_NBR” and “END_REF_MARKER_DISP”
    c. Place END as the last column in the worksheet
    d. Move SIGNED_HIGHWAY_RDBD_ID so that it is before “NUMBER_THRU_LANES”

Signed_ID	Lanes	Surface	Shoulder_R	Shoulder_L	END

4. Place the cursor in the cell just under Signed_ID and run the macro. When prompted enter 6.
    a. When the program is complicated you need to replace the columns "END_REF_MARKER_NBR" and "END_REF_MARKER_DISP" with the values from "END".
    b. The quickest way to do this is to use the TRUNC command in excel to create a column for the "END_REF_MARKER_NBR"
    c. "END" - "END_REF_MARKER_NBR" will give you the values for "END_REF_MARKER_DISP"
    d. Once the values have been replaced, Delete "END"
5. Calculate the lane widths as the roadway with minus the shoulders divided by the number of lanes
6. Run macro "Drop_last_letter_PMIS_ID"
    a. This macro creates a new column that is the PMIS ID without the direction code
    b. This is used later to determine the distance from origin (DFO) using the Texas Reference Marker System (TRM)
7. From the Sample File included with these instructions, copy the TRM sheet
    a. Label it TRM
8. Run the macro "Search_to_match_markers"
9. From the Sample File included with these instruction, copy the "lane_table" sheet
10. Create a new sheet with the following columns copied into it
    a. Number of Lanes, Lane Width, R_Shoulder, L_Shoulder
    b. Place the cursor on the first value in Lanes
    c. Calculate the Capacity by running the macro "fwork"
11. Copy the Capacity Calculations into the PMIS data
12. Use the Following Column Headings (Paste Special, Transpose)

Column Heading
F_YEAR
RES_DIST
C_NBR
PMIS_TYPE
PMIS_ID
NEW_HIWAY_ID
BEG_REF_NBR
BEG_REF_DISP
END_REF_NBR
END_REF_DISP
ST_Maker_DFO
END_Maker_DFO
N_LANES
TOTL_SURF_RDWY

S_RIGHT
S_LEFT
L_WIDTH
CAPACITY
Start
End
Name

13. The value in the column “Name” is just the number of the roadway.
14. Start and End column value need to be calculated use the DFO values and reported in feet.
    a. If the Highway travels W to E set the 0 value to smallest marker value
    b. If the Highway travels S to N set the 0 value to the largest marker value
15. Open Access and import the Excel spreadsheet with all of the data on it.
16. Export the Data as a Database IV file
17. Add the DBIV file in GIS using the Add Data Command

Creating Routes in ArcGIS

1. Add the Census Road tracts using the Add Data command
2. Use the “Select by Attributes” command to select a single highway direction
3. Create a new layer consisting of only that roadway
4. Using the Liner Referencing Toolbox, Create Route Command to create a route using “Name” as the route identifier
5. Go to Tools, Add Route Event
    a. Select the Route and the Data table
    b. Name is the route identifier for both options
    c. Choose line event
    d. Select beginning as “Start”
    e. Select ending as “Finish”
6. Repeat Route process for all roadways

Instructions A.3 Instructions for Installing and Using the Intersection Capacity Tool (written for TxDOT) (contingent upon having the DVD with files)

1. Make sure to open DVD in a DVD compatible drive.

2. It may be best to copy the folder from the DVD to your computer's hard drive.

3. You must have ArcMap installed on your computer to see the signal and roadway layout.

4. To view signal locations along with roadways in the Dallas district, open the folder Data, then the folder Signals, then open the ArcMap document named Signals.

5. If red exclamation points appear next to the layers in the left hand side of your screen, you need to set the data sources of the 3 layers. Do this by first right-clicking on COMBINATION_SIGNALS layer on the left-hand side of your screen. Then select Properties. Select the Source tab. Hit the Set Data Source button. Open the Data folder, then Signals folder, then open SignalDB.gdb. This geodatabase contains the references for the 3 layers. Set each one according to name. You should now be able to see all 3 layers in ArcMap.

6. All of the signals are offset from the roadways because we are currently working with NCTCOG's Demand database. It is known that these roadways are in the wrong geospatial location. The Census roads are in the correct location. When we have completed determining the capacities of the intersections and frontage roads, we will be merging this information with the Census roads so everything lines up properly.

Optional

1. We have developed a tool for calculating the capacities of roads at intersections. This tool is available for download from the website http://www.cs.utexas.edu/~apollo/ArcGIS/. Follow the instructions at this website for installation of the tool into ArcMap. You may need administrator authority to install this on your computer.

2. **Note: This step is what we are currently working on and it may or may not need improvement.**

The previous website also has brief instructions on how to use this tool. Use the Select Features button to select the legs of the intersection and its associated signal. Then hit the CTR Tool button and select CapacityIntersection. A new window will open displaying street name, AM and PM Volume, and Number of Lanes, among other things. This tool will calculate the capacity of each leg of the intersection. If it is a diamond interchange there is a little box in the upper right hand corner which you can check off and then hit Recompute to get new capacities.

You can alter any of the numbers you want in this window. The one most likely in need of change may be the Number of Lanes. Sometimes this is listed as 0 for some roadways. When there is incomplete information about the number of lanes, you can hit the Google Earth Click button and select the point you want to look at in ArcMap. You must have internet access for this feature to work. A new window will open taking you to that exact point in Google Earth. Here you can verify the number of lanes at the intersection you are viewing. Another problem we have noticed is that sometimes a road will not have any volume attributed to it. This will cause the calculated capacity to be 0. We are still deciding how to approach this problem.

You can also eliminate any unwanted legs by double-clicking that specific row in the gray colored far left hand column.

By hitting the Confirm button at the bottom of the table, you will automatically assign the new capacity numbers to the Demand attribute table. In this case, that is the Conformity_PMIS_Dallas layer. Hitting the Cancel button will cause no changes to occur.

Program A.1 DominateForm (Dongliang Xu)
This program is used in the intersection capacity calculation tool.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using System.IO;

using ESRI.ArcGIS.DataSourcesFile;
using ESRI.ArcGIS.esriSystem;
using ESRI.ArcGIS.Framework;
using ESRI.ArcGIS.ArcMapUI;
using ESRI.ArcGIS.Carto;
using ESRI.ArcGIS.Geometry;
using ESRI.ArcGIS.Catalog;
using ESRI.ArcGIS.CatalogUI;
using ESRI.ArcGIS.Location;
using ESRI.ArcGIS.LocationUI;
using ESRI.ArcGIS.Geodatabase;
using ESRI.ArcGIS.Display;


using ESRI.ArcGIS.DataSourcesGDB;
using RKLib.ExportData;

namespace CTR.CTRToolUI
{
    public partial class DominateForm : Form
    {
        private IApplication m_application;
        private IMxDocument doc;
        private IMap map;

        int signal_ID;
        int numLegs;
        float sumAMVolume = 0;
        float sumPMVolume = 0;
        float signalX, signalY;
        private GoogleEarthForm mapForm = null;

        DataTable table;

        #region Public members

        public DominateForm()
        {
            InitializeComponent();
        }
```

```
        ~DominateForm()
        {
        }

        // Store ArcMap application as internal member
        public bool Init(IApplication application)
        {
            m_application = application;
            doc = (IMxDocument)m_application.Document;
            map = doc.FocusMap;


            if (CheckSelection() == false) return false;

            dataGridViewSegments.DataSource = table;

            return true;
        }

        public bool CheckSelection()
        {
            int count = doc.FocusMap.SelectionCount;
            IFeatureLayer signalLayer = TableBuilder.GetLayer(ref doc,
"SIGNALS");
            IFeatureSelection signalSel = TableBuilder.GetLayer(ref
doc, "SIGNALS") as IFeatureSelection;
            if (signalSel .SelectionSet.Count < 1)
            {
                MessageBox.Show("You should choose one intersection
first");
                return false;
            }
            if (signalSel.SelectionSet.Count > 1)
            {
                MessageBox.Show("Multiple intersections has been
selected. Please only select one.");
                return false;
            }


            IEnumIDs ids = signalSel.SelectionSet.IDs;
            int objectID = ids.Next();
            IQueryFilter filter = new QueryFilter();
            filter.WhereClause = "OBJECTID_1 =" + objectID.ToString();
            IFeatureCursor cursor = signalLayer.Search(filter, true);
            IFeature curFeature = cursor.NextFeature();
            signalX = (float)(curFeature.Shape.Envelope.XMin +
curFeature.Shape.Envelope.XMax) / 2.0f;
            signalY = (float)(curFeature.Shape.Envelope.YMin +
curFeature.Shape.Envelope.YMax) / 2.0f;

            signal_ID =
Convert.ToInt32(curFeature.get_Value(curFeature.Fields.FindField("Signa
l_ID")));
```

```
            textBoxSignalID.Text =
curFeature.get_Value(curFeature.Fields.FindField("Signal_ID")).ToString
();
            textBoxStreetA.Text =
curFeature.get_Value(curFeature.Fields.FindField("Street_A")).ToString(
);
            textBoxStreetB.Text =
curFeature.get_Value(curFeature.Fields.FindField("Street_B")).ToString(
);

            IFeatureLayer segLayer = TableBuilder.GetLayer(ref doc,
"Dallas");
            IFeatureSelection segmentSel = TableBuilder.GetLayer(ref
doc, "Dallas") as IFeatureSelection;


            table = new DataTable();
            string tempStr = "";
            int tempInt = 0;
            float tempFloat = 0.0f;
            table.Columns.Add("Street", tempStr.GetType() );
            table.Columns.Add("OBJECTID_12", tempInt.GetType());

            table.Columns.Add("AMHRVOL_AB", tempFloat.GetType());
            table.Columns.Add("AMHRVOL_BA", tempFloat.GetType());
            table.Columns.Add("PMHRVOL_AB", tempFloat.GetType());
            table.Columns.Add("PMHRVOL_BA", tempFloat.GetType());
            table.Columns.Add("DIREC", tempStr.GetType());
            table.Columns.Add("NumLanes", tempInt.GetType());
            table.Columns.Add("MaxAMVolume", tempFloat.GetType());
            table.Columns.Add("MaxPMVolume", tempFloat.GetType());
            table.Columns.Add("g_iOverC", tempFloat.GetType());
            table.Columns.Add("s", tempFloat.GetType());
            table.Columns.Add("Capacity", tempInt.GetType());

            ids = segmentSel.SelectionSet.IDs;
            while (true)
            {
                objectID = ids.Next();
                if (objectID < 0) break;

                filter = new QueryFilter();
                filter.WhereClause = "OBJECTID_12 =" +
objectID.ToString();
                cursor = segLayer.Search(filter, true);
                curFeature = cursor.NextFeature();

                DataRow row = table.NewRow();

                row["Street"] =
curFeature.get_Value(curFeature.Fields.FindField("Street"));
                row["OBJECTID_12"] = objectID;

                float maxAM = 0.0f;
```

```
                float maxPM = 0.0f;
                row["NumLanes"] =
curFeature.get_Value(curFeature.Fields.FindField("NUMBER_THR"));
                row["AMHRVOL_AB"] =
curFeature.get_Value(curFeature.Fields.FindField("AMHRVOL_AB"));
                row["AMHRVOL_BA"] =
curFeature.get_Value(curFeature.Fields.FindField("AMHRVOL_BA"));
                row["PMHRVOL_AB"] =
curFeature.get_Value(curFeature.Fields.FindField("PMHRVOL_AB"));
                row["PMHRVOL_BA"] =
curFeature.get_Value(curFeature.Fields.FindField("PMHRVOL_BA"));
                maxAM = Convert.ToDouble(row["AMHRVOL_AB"]) > maxAM ?
(float)(row["AMHRVOL_AB"]) : maxAM;
                maxAM = Convert.ToDouble(row["AMHRVOL_BA"]) > maxAM ?
(float)(row["AMHRVOL_BA"]) : maxAM;
                maxPM = Convert.ToDouble(row["PMHRVOL_AB"]) > maxPM ?
(float)(row["PMHRVOL_AB"]) : maxPM;
                maxPM = Convert.ToDouble(row["PMHRVOL_BA"]) > maxPM ?
(float)(row["PMHRVOL_BA"]) : maxPM;
                row["MaxAMVolume"] = maxAM;
                row["MaxPMVolume"] = maxPM;
                row["DIREC"] =
curFeature.get_Value(curFeature.Fields.FindField("DIREC"));

                table.Rows.Add(row);
            }


            Compute();

            return true;
        }

        #endregion

        private void DominateForm_Load(object sender, EventArgs e)
        {

        }
        void Compute()
        {
            sumAMVolume = 0;
            sumPMVolume = 0;
            numLegs = table.Rows.Count;
            foreach (DataRow row in table.Rows)
            {
                sumAMVolume += Convert.ToInt32(row["MaxAMVolume"]);
                sumPMVolume += Convert.ToInt32(row["MaxPMVolume"]);
            }

            string columnName;
            float sumVolume = 0;
            if (sumAMVolume > sumPMVolume)
            {
```

```
                sumVolume = sumAMVolume;
                columnName = "MaxAMVolume";
            }
            else
            {
                sumVolume = sumPMVolume;
                columnName = "MaxPMVolume";
            }

            float constantC = 100;
            if (checkBoxDiamond.Checked == true)
            {
                constantC = 150;
            }

            textBoxNumLegs.Text = numLegs.ToString();
            textBoxSumVolume.Text = sumVolume.ToString();
            textBoxC.Text = constantC.ToString();

            foreach (DataRow row in table.Rows)
            {
                float g_iOverC = constantC - numLegs * 4;
                g_iOverC = g_iOverC * ((float)row[columnName] /
sumVolume);
                g_iOverC = g_iOverC / constantC;
                row["g_iOverC"] = g_iOverC;
                row["s"] = 1900 * Convert.ToInt32(row["NumLanes"]) *
0.95f * 0.95f * 0.90f;
                row["Capacity"] = (int)((float)(row["g_iOverC"]) *
(float)(row["s"]));
            }
        }
        private void buttonConfirm_Click(object sender, EventArgs e)
        {
            IFeatureLayer segLayer = TableBuilder.GetLayer(ref doc,
"Dallas");
             IFeatureLayer signalLayer = TableBuilder.GetLayer(ref doc,
"SIGNALS");
            IFeatureSelection segmentSel = TableBuilder.GetLayer(ref
doc, "Dallas") as IFeatureSelection;
            IEnumIDs ids = segmentSel.SelectionSet.IDs;

            while (true)
            {
                int objectID = ids.Next();
                if (objectID < 0) break;

                IQueryFilter filter = new QueryFilter();
                filter.WhereClause = "OBJECTID_12 =" +
objectID.ToString();
                IFeatureCursor cursor = segLayer.Search(filter, true);
                IFeature curFeature = cursor.NextFeature();
```

```
                DataRow specifiedRow = null;
                foreach (DataRow row in table.Rows)
                {
                    if (Convert.ToInt32(row["OBJECTID_12"]) ==
objectID)
                    {
                        specifiedRow = row;
                        break;
                    }
                }
                if (specifiedRow == null) continue;


curFeature.set_Value(curFeature.Fields.FindField("Intersection_Capacity
"), Convert.ToDouble(specifiedRow["Capacity"]));

curFeature.set_Value(curFeature.Fields.FindField("NUMBER_THR"),
Convert.ToDouble(specifiedRow["NumLanes"]));

curFeature.set_Value(curFeature.Fields.FindField("Signal_ID"),
signal_ID);
                curFeature.Store();
            }

            this.Hide();
        }

        private void buttonCancel_Click(object sender, EventArgs e)
        {
            this.Hide();
        }

        private void label1_Click(object sender, EventArgs e)
        {

        }

        private void checkBoxDiamond_CheckedChanged(object sender,
EventArgs e)
        {
            Compute();
        }

        private void textBoxNumLanes_TextChanged(object sender,
EventArgs e)
        {

        }

        private void buttonRecompute_Click(object sender, EventArgs e)
        {
            Compute();
        }
```

```
        private void
dataGridViewSegments_RowHeaderMouseDoubleClick(object sender,
DataGridViewCellMouseEventArgs e)
        {
            table.Rows.RemoveAt(e.RowIndex);
            Compute();
        }

        private void buttonDefaultInfo_Click(object sender, EventArgs
e)
        {
            DefaultInfoForm form = new DefaultInfoForm();
            form.Show();
        }

        private void buttonGoogleMap_Click(object sender, EventArgs e)
        {
            IPoint point = new ESRI.ArcGIS.Geometry.Point();
            point.X = signalX;
            point.Y = signalY;

            if (mapForm == null)
            {
                // Create Routing Form
                mapForm = new GoogleEarthForm();
                mapForm.Init(m_application);

                mapForm.SetPoint(point);
                // show form
                mapForm.Show();

                // Set ArcMap window as owner for Routing Form
                SetWindowLong(mapForm.Handle.ToInt32(), GWL_HWNDPARENT,
m_application.hWnd);
            }
            else
            {
                // just show/hide form
                if (!mapForm.Visible)
                {
                    mapForm.Redraw();
                    mapForm.SetPoint(point);
                }
            }
        }



        #region Imported functions

        // Needed to show non-modal Routing form
        [System.Runtime.InteropServices.DllImport("user32", EntryPoint
= "SetWindowLongA", ExactSpelling = true, CharSet =
System.Runtime.InteropServices.CharSet.Ansi, SetLastError = true)]
```

```
        public static extern int SetWindowLong(int hwnd, int nIndex,
int dwNewLong);
        public const int GWL_HWNDPARENT = (-8);

        #endregion


    }
}
```

Program A.2 GoogleEarthForm (Dongliang Xu)
This program is used in the Google Earth Click tool.

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.Reflection;
using System.IO;
using System.Runtime.InteropServices;
using System.Diagnostics;

using EARTHLib;
using ESRI.ArcGIS.DataSourcesFile;
using ESRI.ArcGIS.esriSystem;
using ESRI.ArcGIS.Framework;
using ESRI.ArcGIS.ArcMapUI;
using ESRI.ArcGIS.Carto;
using ESRI.ArcGIS.Geometry;
using ESRI.ArcGIS.Catalog;
using ESRI.ArcGIS.CatalogUI;
using ESRI.ArcGIS.Location;
using ESRI.ArcGIS.LocationUI;
using ESRI.ArcGIS.Geodatabase;
using ESRI.ArcGIS.Display;

namespace CTR.CTRToolUI
{
    public partial class GoogleEarthForm : Form
    {
        #region Private Memebers
        // ArcMap application
        private IApplication m_application;

        public ApplicationGE googleEarth;
        private IntPtr GEHrender = (IntPtr)0;
        private IntPtr GEHmain = (IntPtr)0;
        private IntPtr GEParentHrender = (IntPtr)0;

        #endregion

        public GoogleEarthForm()
        {
            InitializeComponent();

            InitMap();
        }
        ~GoogleEarthForm()
        {
        }
```

```
        public void DestroyGoogleEarth()
        {
            if (GEHmain != (IntPtr)0)
            {
                DestroyWindow(GEHrender);
                DestroyWindow(GEParentHrender);
                DestroyWindow(GEHmain);

                KillProcess("googleearth");
            }
        }

        public void Init(IApplication application)
        {
            m_application = application;
        }
        public void SetPoint(IPoint point)
        {
            googleEarth.SetCameraParams(point.Y, point.X, 0,
AltitudeModeGE.RelativeToGroundAltitudeGE, 1000, 45, 0, 0.9);
        }
        public void Redraw()
        {
            GEHmain = (IntPtr)googleEarth.GetMainHwnd();
            GEHrender = (IntPtr)googleEarth.GetRenderHwnd();
            GEParentHrender = GetParent(GEHrender);

            MoveWindow(GEHrender, 0, 0, panelMap.Width,
panelMap.Height, true);
            SetParent(GEHrender, panelMap.Handle);

            //MoveWindow(GEHmain, 0, 0, panelMap.Width,
panelMap.Height, true);
            SetParent(GEHmain, panelHideGE.Handle);
            this.Show();
        }
        public void InitMap()
        {
            googleEarth = new ApplicationGE();

            Redraw();
        }
        private bool KillProcess(string name)
        {
            Process[] processes = Process.GetProcesses();
            foreach (Process clsProcess in processes)
            {
                if (clsProcess.ProcessName.StartsWith(name))
                {
                    clsProcess.Kill();
                    //process killed, return true
                    return true;
                }
            }
```

```
            //process not found, return false
            return false;


        }


        #region Imported functions
        // Needed to show non-modal Routing form
        [System.Runtime.InteropServices.DllImport("user32", EntryPoint
= "SetWindowLongA", ExactSpelling = true, CharSet =
System.Runtime.InteropServices.CharSet.Ansi, SetLastError = true)]
        public static extern int SetWindowLong(int hwnd, int nIndex,
int dwNewLong);
        public const int GWL_HWNDPARENT = (-8);

        [DllImport("user32.dll")]
        static extern bool DestroyWindow(IntPtr hWnd);

        public delegate int EnumWindowsProc(IntPtr hwnd, int lParam);

        [DllImport("user32", CharSet = CharSet.Auto)]
        public extern static IntPtr GetParent(IntPtr hWnd);

        [DllImport("user32", CharSet = CharSet.Auto)]
        public extern static bool MoveWindow(IntPtr hWnd, int X, int Y,
int nWidth, int nHeight, bool bRepaint);

        [DllImport("user32", CharSet = CharSet.Auto)]
        public extern static IntPtr SetParent(IntPtr hWndChild, IntPtr
hWndNewParent);
        #endregion

        private void buttonHide_Click(object sender, EventArgs e)
        {
            this.Hide();
        }


    }
}
```

Program A.3 Capacity Calculation Excel Macro (Spurgeon)

```
Function get_cell(ByRef row As Long, ByRef col As Long, ByRef val As Variant)
'precondition: ONE cell is selected
'postcondition: cell address and value will be passed to row and col accordingly
   row = ActiveCell.row
   col = ActiveCell.Column
   val = ActiveCell.Value
End Function

Function write_Out(ByVal cell_val As Double, ByRef row As Long)
   ActiveSheet.Cells(row, 10).Value = cell_val
End Function
Function data_size(ByRef row_count As Integer)
'precondition: select the first cell of the column which the data size needs to be
determined
'postcondition: the number of data in the ENTIRE column will be set to row_count and
the original cell is selected
   Dim temp As Double
   Dim col1 As Long, row1 As Long, val1 As Variant
   row_count = 0
   Do While ActiveCell.Value <> vbNullString
      row_count = row_count + 1
      ActiveCell.Offset(1, 0).Select
   Loop
   temp = row_count * -1
   ActiveCell.Offset(temp, 0).Select
End Function

Function search_lane_table(ByRef sh_right As Long, ByRef sh_left As Long, la_width
As Long, _
num_lane As Long, sh_crit As Long, sh_factor As Double)
'precondition: the shoulder left, shoulder right, lane and width and number of lanes must
be passed intot the function
'postcondtion: the correction factor of shoulder width

   If sh_right >= sh_left Then
      sh_crit = sh_left
   Else
   sh_crit = sh_right
   End If
'Determing which shoulder with in the critical shoulder width

   If num_lane < 3 Then
```

```
    Sheets("lane_table").Activate
    ActiveSheet.Cells(2, 1).Select
    End If

    If num_lane < 3 And sh_right < 6 And sh_left < 6 Then
    Sheets("lane_table").Activate
    ActiveSheet.Cells(10, 1).Select
    End If

    If num_lane >= 3 Then
    Sheets("lane_table").Activate
    ActiveSheet.Cells(18, 1).Select
    End If

    If num_lane >= 3 And sh_right < 6 And sh_left < 6 Then
    Sheets("lane_table").Activate
    ActiveSheet.Cells(26, 1).Select
    End If

    Do While sh_crit < ActiveCell.Value
       ActiveCell.Offset(1, 0).Select
    Loop

    Dim RowCount As Long
    RowCount = ActiveCell.row

    ActiveSheet.Cells(1, 2).Select

    Do While la_width < ActiveCell.Value
       ActiveCell.Offset(0, 1).Select
    Loop

    Dim Colcount As Long
    Colcount = ActiveCell.Column

    ActiveSheet.Cells(RowCount, Colcount).Select

    sh_factor = ActiveCell.Value


End Function
Sub fwork()
Dim row As Long, col As Long, val As Long, row_count As Integer, i As Long
```

```
Dim sh_right As Long, sh_left As Long, la_width As Long, num_lane As Long, sh_crit
As Long
Dim sh_factor As Double
Dim temprow As Long, tempcol As Long


i = 0
row = 0
col = 0
val = 0
row_count = 0

    Call data_size(row_count)

    Call get_cell(row, col, val)

    For i = 2 To row_count + 1
        temprow = row
        tempcol = col


         num_lane = ActiveCell.Value
         ActiveCell.Offset(0, 1).Select

         sh_right = ActiveCell.Value
         ActiveCell.Offset(0, 1).Select

         sh_left = ActiveCell.Value
         ActiveCell.Offset(0, 1).Select

         la_width = ActiveCell.Value
         ActiveCell.Offset(0, 1).Select

         sh_crit = 0
         sh_factor = 0

         Call search_lane_table(sh_right, sh_left, la_width, num_lane, sh_crit, sh_factor)

         Sheets("Sheet2").Activate
         ActiveSheet.Cells(temprow, tempcol).Select

         val = 2000 * val * sh_factor * 0.9 * 1

        Call write_Out(val, i)
```

```
        ActiveCell.Offset(1, 0).Select
        Call get_cell(row, col, val)
    Next i

End Sub
```

Program A.4 Rui Gao's Programs

For the code of all other programs mentioned in this thesis, please see:

Gao, R. *Programming for Enhancing System-Wide Traffic Operations - The Design of a GIS Based Traffic Control Planning Tool*. Thesis. The University of Texas at Austin, 2008. Print.

Works Cited

Critical Issues in Transportation. Rep. Washington DC: Transportation Research Board, 2006. Print.

Dallas - Fort Worth Regional Travel Model (DFWRTM): Model Description. Rep. North Central Texas Council of Governments Transportation Department Travel Model Development Group, Sept. 2009. Web. Oct. & Nov. 2009. <http://www.nctcog.org/trans/modeling/documentation/DFWRTMModelDescription.pdf>.

Domencich, T., and D. L. McFadden. *Urban Travel Demand: A Behavioral Analysis*. North-Holland Co., 1975. Print.

Gao, R. *Programming for Enhancing System-Wide Traffic Operations - The Design of a GIS Based Traffic Control Planning Tool*. Thesis. The University of Texas at Austin, 2008. Print.

Highway Capacity Manual 2000. Detroit: Transportation Research Board, 2000. Print.

Lomax, T., and D. Schrank. *The 2009 Urban Mobility Report*. Rep. College Station: Texas Transportation Institute, 2009. Print.

Longley, P. A., M. F. Goodchild, D. J. Maguire, and D. W. Rhind. *Geographic Information Systems and Science*. New York: John Wiley & Sons, 2001. Print.

Manual on Uniform Traffic Control Devices (MUTCD) - FHWA. Web. Nov. 2009. <http://mutcd.fhwa.dot.gov/>.

Ort, J. D., and L. G. Willumsen. *Modelling Transport*. New York: Wiley, 2001. Print.

Roess, R. P. *Traffic Engineering*. Upper Saddle River, N.J: Pearson/Prentice Hall, 2004. Print.

Spurgeon, E. C. *Improving Traffic Operations through the Creation of a GIS Based Traffic Control Planning Tool*. Thesis. The University of Texas at Austin, 2008. Print.

Tomlinson, R. *History of Geographic Information Systems: Perspectives from the Pioneers*. Ed. T. W. Foresman. Upper Saddle River, NJ: Prentice Hall PTR, 1998. Print.

"TransCAD Transportation Planning Software Overview." *Caliper - Mapping software, GIS, and transportation planning software*. Web. Nov. 2009. <http://www.caliper.com/tcovu.htm>.

"Universal Detector Data Archive - DalTrans." *Texas Transportation Institute (TTI): Research. Service. Results.* Web. Nov. 2009. <http://ttidallas.tamu.edu/detectordataarchive/daltrans/default.htm>.

Zhang, Z., M. R. Rechtien, D. W. Fowler, and W. R. Hudson. *A Summary of Pavement and Material-Related Databases Within the Texas Department of Transportation*. Tech. Austin: Center for Transportation Research, 1999. *A Summary of Pavement and Material-Related Databases Within the Texas Department of Transportation*. Texas Department of Transportation, Sept. 1999. Web. Oct. & nov. 2009. <http://www.utexas.edu/research/ctr/pdf_reports/1785_1.pdf>.

Vita

Andrew W. Karl was born on September 3, 1986 in White Plains, NY. He is the son of William and Andrea Karl. He has a younger sister, Julia. After graduating from John F. Kennedy Catholic High School in 2004, he attended Manhattan College in Riverdale, NY. He received a Bachelor of Science in Civil Engineering in May 2008. In August 2008 he entered the Graduate School at The University of Texas at Austin. After receiving a Master of Science in Engineering he plans to work in the field of Transportation Engineering.

Permanent Address: 12 Summit Avenue
Putnam Valley, NY 10579

This thesis was typed by Andrew W. Karl.